

Personal DECstation /DECsystem 5000 Series

Maintenance Guide

First printing, February 1992
Second printing, June 1992
Third printing, April 1993

© Digital Equipment Corporation 1991.

U.S.A.

This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio frequency interference. Operation of this equipment in a residential area may cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

The following are trademarks of Digital Equipment Corporation:

DEC	MicroVMS	VAX
DECnet	PDP	VAXBI
DECstation	ThinWire	VAXcluster
DECsystem	TURBOchannel	VAXstation
DECUS	ULTRIX	VMS
DESTA	ULTRIX-32	VT
MicroVAX	UNIBUS	digital ™

Contents

About This Guide	xxiii
-------------------------------	--------------

Part I System Overview

1 Base System

System Hardware Configurations	1-2
System Unit	1-5
Controls and Indicators	1-7
External System Unit Connectors	1-8
Internal Base System Module Connectors	1-9
Clear-NVR Jumper	1-11
System Boot ROM	1-12
For Further Information	1-13

2 Hardware Options and Peripherals

CPU Module	2-1
Memory Modules	2-2
Monitors and Terminals	2-2
Keyboard and Mouse	2-3
SCSI Drives	2-4
TURBOchannel Option Modules	2-4
For Further Information	2-5

3 A Server Configuration

Introduction	3-1
Connecting a Console Terminal.....	3-2
Halting a server	3-3
Default Settings for Using the Console Terminal.....	3-4
Operational background information	3-4

Part II Troubleshooting

4 Troubleshooting Overview

Troubleshooting Overview	4-2
Four Troubleshooting Techniques	4-2
Reference Material	4-4
General Troubleshooting Tips	4-5

5 The Powerup Self-test

Troubleshooting Techniques.....	5-2
Invoking the Powerup Self-Test	5-3
Interpreting Video Error Messages.....	5-4
Example: Power-up Selftest Failure	5-5
Connecting and Using an Alternate Terminal	5-7
Minimum Configuration Troubleshooting With the System LED.....	5-9
Understanding the Power-up Self-Test Sequence and the LED Display	5-10
The Power-up Sequence	5-11
What to do when the Power-up LED Indicator is Inactive ..	5-15
How to Troubleshoot the R4000 CPU Module LEDs	5-16

6 ROM-Based Console and Diagnostics

Console Commands.....	6-2
Similarity to Other DS5000 Systems' Console Commands	6-4
On-line help	6-5

Configuration Display	6-6
Example: cnfg Command output	6-6
The cnfg <i>x</i> Command	6-8
Example: cnfg 3 Output Screen	6-9
TURBOchannel Slot Number Assignments	6-11
Environment Variables	6-12
Interpreting Error Messages	6-14
Console command error messages	6-17
Console exception messages	6-18
Diagnostic Tests	6-19
Selecting diagnostic routines	6-21

7 ULTRIX Error Reporter and System Exerciser

Troubleshooting Techniques	7-1
ULTRIX Error Logs	7-2
Generating error reports	7-2
On-line help	7-4
ULTRIX error log event types	7-5
Interpreting part one of the error report	7-5
Sample error report	7-6
Interpreting part two of the error report	7-7
Memory Error Report	7-8
Analyzing memory error report	7-8
Error and status register error logs	7-9
Error and status register error log: Example 1	7-10
Error and status register error log: Example 2	7-12
System Exercisers	7-14
On-line help	7-15
Running the system exercisers	7-16
Running several exercisers at once	7-16
Running individual exercisers	7-16
Exercising the file system and memory	7-16
Exercising local file systems	7-17
Exercising the system memory	7-17
Exercising shared memory	7-18
Exercising peripherals	7-18
Exercising disk drives	7-19
Exercising tape drives	7-20
The mtx tape drive exerciser	7-20
The tapex tape drive exerciser	7-21

Exercising the network and communication systems	7-22
Exercising the TCP/IP network	7-22
Exercising the terminal communications	7-23
Exercising line printers	7-24

8 Obtaining a Crash Dump

Troubleshooting Techniques	8-2
Verifying the system is hung	8-2
Crashing the system	8-3
Creating a dump file	8-4
Setting the haltaction Environmental Variables	8-4

9 Procedural Flow Charts

No Display at Power-Up	9-1
Missing SCSI Device	9-3
Missing TURBOchannel Option	9-5
Missing Memory SIMMs	9-7
Missing SERIAL.bus Device	9-9

Part III Removing and Replacing FRUs

10 Base System Hardware

Using the Antistatic Kit	10-2
Removing and Installing the System Unit Cover	10-3
To Remove the System Unit Cover	10-3
To Install the System Unit Cover	10-4
Installing and Removing a Serial Line Adapter	10-5
To install a serial line adapter	10-5
To remove a serial line adapter	10-5
Installing and Removing a SCSI Controller Terminator	10-6
To Install a SCSI Controller Terminator	10-6
To Remove a SCSI Controller Terminator	10-6
Installing and Removing a SCSI Chain Terminator	10-7
To Install a SCSI Chain Terminator	10-7
To Remove a SCSI Chain Terminator	10-7
Installing and Removing a ThickWire Ethernet Loopback Connector	10-8
To Install a ThickWire Ethernet Loopback Connector . . .	10-8

To Remove a ThickWire Ethernet Loopback Connector	10-8
Installing and Removing a Serial Line Loopback Connector	10-9
To Install a Serial Line Loopback Connector	10-9
To Remove a Serial Line Loopback Connector	10-9

11 Storage Devices

SCSI Requirements	11-2
SCSI Cable Lengths	11-3
Internal Drives	11-4
Removing and Installing an Internal Floppy Disk Drive	11-6
To remove a floppy disk drive	11-6
To install a floppy disk drive	11-6
Removing and Installing a Floppy Disk Mounting Bracket	11-10
To remove a floppy disk mounting bracket	11-10
To install a floppy disk mounting bracket	11-10
Removing and Installing a Floppy Port Cover	11-11
To remove the floppy port cover	11-11
To install a floppy port cover	11-11
Removing and Installing an Internal Hard Disk Drive	11-12
To remove an internal hard disk drive	11-12
To install an internal hard disk drive	11-13
ID Settings for Internal Drives	11-14
Setting SCSI IDs	11-15
Setting the SCSI ID on an RZ23L hard disk drive	11-15
Setting the SCSI ID on an RZ24 hard disk drive	11-17
Setting the SCSI ID on an RZ24L hard disk drive	11-19
Setting the SCSI ID on an RZ25 hard disk drive	11-20
Setting the SCSI ID on an RZ26 hard disk drive	11-23
Setting the SCSI ID on an TLZ06 tape drive	11-25
Setting the drive select switch on an RX26 floppy disk drive	11-26
For Further Information	11-27

12 CPU, TURBOchannel Options, and Memory

Removing and Replacing a CPU Module	12-2
To Remove a CPU Module	12-5
To Install a CPU Module	12-5
Removing and Installing Memory Modules	12-7
Identifying Memory Modules	12-7
Memory Slot Numbers and Address Ranges	12-8
To Remove a Memory Module	12-11
To Install a Memory Module	12-13
Removing and Installing TURBOchannel Option Modules	12-15
To Remove an Option Module	12-15
To Install an Option Module	12-15
Ethernet Option Module ESAR Chip	12-17
Base System ESAR Chip	12-19
To Remove the Base System ESAR Chip	12-19
To Install the Base System ESAR Chip	12-19
Removing and Installing the Base System Module	12-21
To Remove the Base System Module	12-23
To Install the Base System Module	12-25
Installing and Removing the Power Supply	12-26
To Remove the Power Supply	12-26
To Install the Power Supply	12-27
For Further Information	12-29

13 Keyboard and Pointing Devices

Keyboard	13-2
Mouse	13-3
Removing and Installing the Input Devices	13-4
To Disconnect the Keyboard and Mouse	13-4
To Install the Keyboard and Mouse	13-4

Part IV Appendices

A Equipment Specifications

Personal DECstation 5000 System Unit Equipment Specifications	A-2
VRC16 Monitor Equipment Specifications	A-4
VRM17 Monitor Equipment Specifications	A-6
LK501-AA Keyboard Equipment Specifications	A-9
VSXXX-BB Mouse Equipment Specifications	A-11
RZ23L SCSI Hard Disk Drive Equipment Specifications ...	A-13
RZ24L SCSI Hard Disk Drive Equipment Specifications ...	A-15
RZ25 SCSI Hard Disk Drive Equipment Specifications	A-17
RZ26 SCSI Hard Disk Drive Equipment Specifications	A-19
TLZ06 Cassette Tape Drive Equipment Specifications	A-21
RX26 Diskette Drive Equipment Specifications	A-22

B Part Numbers

Basic System Components	B-2
SCSI and Floppy Disk Hardware	B-3
Monitors	B-4
Input Devices	B-4
Cables and Cords	B-5
Miscellaneous Hardware	B-5
Connectors, Adapters, and Terminators	B-6
Hardware Documentation	B-7
Software Documentation	B-8

C Connector Pin Assignments

SCSI Connector Pin Assignments	C-2
ThickWire Ethernet Connector Pin Assignments	C-3
EIA RS232 Communications Connectors Pin Assignments	C-4
Power Supply Pin Assignments	C-6
ISDN Connector Pin Assignments	C-7
SERIAL.bus	C-8
Sound Connector	C-9
Loopback Connector Pin Assignments	C-10

D Console Commands

Entering Console Commands	D-2
Rules for entering console commands	D-2
Conventions used in this chapter	D-3
Some terms used in this chapter	D-3
Introduction to the Command Reference	D-4
Console command format summary	D-5
? Command	D-7
boot Command	D-8
Important information about the boot command	D-10
cat Command	D-11
cnfg Command	D-12
General System Configuration Displays	D-12
Individual Configuration Displays	D-14
Base system configuration displays	D-14
Ethernet controller configuration displays	D-16
SCSI controller displays	D-16
3D graphics module configuration displays	D-18
Color frame buffer graphics module configuration displays	D-19
2D graphics accelerator module configuration displays	D-20
d Command	D-21
e Command	D-23
erl Command	D-25
go Command	D-25
init Command	D-26
ls Command	D-27
passwd Command	D-28
printenv Command	D-29
restart Command	D-29
script Command	D-30
setenv Command	D-31
sh Command	D-35
t Command	D-36
test Command	D-36
unsetenv Command	D-37
Console Command Error Messages	D-38

E Base System Test Commands and Messages

Locating Individual Tests in This Appendix	E-2
Tests	E-6
ab/caps—Desktop Bus Device Capabilities Test	E-6
Desktop bus device capabilities test error messages	E-6
ab/cntl—Desktop Bus Controller Test	E-7
ab/ctrs—Desktop Bus Counters Test	E-7
Desktop bus counters test error message	E-8
ab/nm—Display Next Message Test	E-8
ab/t—Desktop Bus Tables Verification Tool	E-8
cache/data—Cache Data Test	E-8
Cache data test error messages	E-9
cache/fill—Cache Fill Test	E-10
Cache fill test error messages	E-10
cache/isol—Cache Isolate Test	E-11
Cache isolate test error messages	E-11
cache/reload—Cache Reload Test	E-12
Cache reload test error messages	E-13
cache/seg—Cache Segment Test	E-14
Cache segment test error messages	E-14
scache/data - Secondary Cache Test	E-15
Scache Error	E-15
crt/ascii—ASCII Font Test	E-16
crt/cursmem—Cursor Memory Test	E-16
Cursor memory test error messages	E-16
crt/e—‘E’ Test	E-16
crt/fb—Frame Buffer Test	E-17
Frame buffer test error messages	E-17
crt/h—‘H’ Test	E-17
crt/init—Chip Initialization Test	E-18
crt/patterns—Screen Patterns Test	E-18
crt/vdac—Vdac/Color Palette Test	E-18
Vdac/color palette test error message	E-18
epm/dump—EEPROM Contents Display Test	E-19
epm/init—EEPROM Reinitialization Test	E-19
epm/sign—EEPROM Signature Data Check Test	E-19
EEPROM signature data check error messages	E-19
epm/write—EEPROM Read/Modify/Write Test	E-20
fctr/intvl—Free-running Counter Interval Test	E-20
Free-running counter interval test error messages	E-20
fctr/run—Free-Running Counter Increment Test	E-21
Free-running counter increment test error message	E-21
fdi/dor—Floppy DOR Register Test	E-21
Floppy DOR register test error messages	E-22
fdi/exfdc—Exercise Floppy Disk Controller Test	E-22

Exercise floppy disk controller test error messages	E-23
fdi/read—Floppy Media Read Test	E-24
Floppy media read test error messages	E-24
fdi/reset—Reset Floppy Disk Interface Test	E-24
Reset floppy disk interface test error messages	E-25
fdi/t—DOR Subtest	E-25
fdi/wrtrd—Write/Read Floppy Media Test	E-26
Write/read floppy media test error messages	E-26
fpu—Floating-Point Unit (FPU) Test	E-27
FPU test error messages	E-27
isdn/audio—ISDN Controller Audio Test	E-28
isdn/base—ISDN Baseline Test	E-29
ISDN baseline test error messages	E-29
isdn/dma—ISDN Controller Peripheral Port Test	E-30
ISDN controller peripheral port test error message	E-30
mem/float10—Floating 1/0 Memory Test	E-30
Floating 1/0 memory test error messages	E-31
mem—Memory Module Test	E-31
Memory module test error messages	E-31
mem/init—Zero Memory Utility	E-32
mem/select—RAM Select Lines Test	E-32
RAM select lines test error messages	E-33
mfg/done—DIAGDN Bit Test	E-33
misc/cpu-type—CPU-Type Utility	E-33
CPU-type utility message	E-33
misc/wbpart—Partial Write Test	E-34
Partial write test error messages	E-34
ni/cllsn—Collision Test	E-34
Collision test error messages	E-35
ni/common—Common Diagnostic Utilities	E-35
Common diagnostic utility error messages	E-35
ni/crc—Cyclic Redundancy Code (CRC) Test	E-36
CRC test error messages	E-36
ni/ctrs—Display MOP Counters Utility	E-37
ni/dma1—Ethernet-Direct Memory Access (DMA) Registers Test	E-38
Ethernet-DMA registers test error messages	E-38
ni/dma2—Ethernet-Direct Memory Access (DMA) Transfer Test	E-39
Ethernet-DMA transfer test error messages	E-39
ni/esar—Ethernet Station Address ROM (ESAR) Test	E-40
ESAR test error messages	E-41
ni/ext-lb—External Loopback Test	E-41
External loopback test error messages	E-42
ni/int—Interrupt Request (IRQ) Test	E-42

IRQ test error messages	E-43
ni/int-lb—Internal Loopback Test	E-43
Internal loopback test error messages	E-44
ni/m-cst—Multicast Test	E-45
Multicast test error messages	E-45
ni/promisc—Promiscuous Mode Test	E-46
Promiscuous mode test error messages	E-46
ni/regs—Registers Test	E-47
Registers test error messages	E-47
ni/setup—LANCE Setup Utilities.	E-48
LANCE-setup utilities error messages	E-49
rtc/nvr—Nonvolatile RAM (NVR) Test	E-49
NVR test error messages	E-49
rtc/period—Real-Time Clock (RTC) Period Test	E-50
RTC period test error messages	E-50
rtc/regs—Real-Time Clock Registers Test	E-51
Real-time clock registers test error messages	E-51
rtc/time—Real-Time Test	E-51
Real-time test error messages	E-52
scc/access—Serial Communication Chip (SCC) Access Test	E-52
SCC access test error message	E-53
scc/dma—Serial Communication Chip (SCC) Direct Memory Access (DMA) Test	E-53
SCC DMA test error messages	E-54
scc/int—Serial Communication Chip (SCC) Interrupts Test	E-55
SCC interrupts test error messages	E-55
scc/io—Serial Communication Chip (SCC) I/O Test	E-56
SCC I/O test error messages	E-56
scc/pins—Serial Communication Chip (SCC) Pins Test	E-57
SCC pins test error messages	E-58
scc/tx-rx—Serial Communication Chip (SCC) Transmit and Receive Test	E-59
SCC transmit and receive test error messages	E-60
scsi/cntl—SCSI Controller Test	E-61
SCSI controller test error messages	E-62
scsi/sdiag—SCSI Send Diagnostics Test	E-62
SCSI send diagnostics test error messages	E-63
scsi/target—SCSI Target Test	E-63
SCSI target test error messages	E-64
tlb/prb—Translation Lookaside Buffer (TLB) Probe Test . . .	E-66
TLB probe test error messages	E-66
tlb/reg—Translation Lookaside Buffer (TLB) Registers Test	E-66

TLB registers test error messages	E-66
Diagnostic LED Codes	E-67

F CPU and System Registers

CPU Registers	F-2
CPU Registers for R3000A Module	F-2
CPU Registers for R4000 Only	F-3
Cause Register R3000 Only	F-3
Cause Register R4000 Only	F-5
Exception Program Counter (EPC) Register R3000 and R4000	F-7
Status Register R3000 Only	F-8
Diagnostic status	F-9
Status Register R4000 Only	F-11
Diagnostic Status Field	F-13
BadVAddr Register R3000 and R4000	F-13
Error Exception Program Counter (Error EPC) R4000 Only	F-14
Configuration Register R4000 Only	F-14
System Registers	F-16
Data Buffers 3 to 0	F-16
System Support Register (SSR)	F-17
SSR<31>	F-18
SSR<30>	F-18
SSR<29>	F-18
SSR<28>	F-18
SSR<27:23>	F-18
SSR<22>	F-18
SSR<21>	F-18
SSR<20>	F-18
SSR<19>	F-18
SSR<18>	F-18
SSR<17>	F-18
SSR<16>	F-18
SSR<15>DIAGDN	F-19
SSR<14:13>	F-19
SSR<12>	F-19
SSR<11>	F-19
SSR<10>	F-19
SSR<9>	F-19
SSR<8>	F-19
SSR<7>	F-19
SSR<6>	F-19

SSR<5>	F-19
SSR<4:1>	F-20
SSR<0>LED	F-20
System Interrupt Register (SIR)	F-20
SIR<31>	F-22
SIR<30>	F-22
SIR<29>	F-22
SIR<28>	F-22
SIR<27>	F-22
SIR<26>	F-23
SIR<25>	F-23
SIR<24>	F-23
SIR<23>	F-23
SIR<22>	F-23
SIR<21>	F-23
SIR<20>	F-24
SIR<19>	F-24
SIR<18>	F-24
SIR<17>	F-24
SIR<16>	F-24
SIR<15>	F-24
SIR<14>UNSCUR	F-24
SIR<13>	F-25
SIR<12>	F-25
SIR<11>	F-25
SIR<10>NRMOD	F-25
SIR<9>	F-25
SIR<8>	F-25
SIR<7>Floppy HDSTATUS	F-25
SIR<6>	F-25
SIR<5>	F-25
SIR<4>Floppy XDSTATUS	F-25
SIR<3>	F-25
SIR<2>	F-25
SIR<1>	F-26
SIR<0>	F-26
System Interrupt Mask Register	F-26
<31:0>	F-26
Memory Error Register (MER)	F-26
MER<31:28>	F-27
MER<27>	F-27
MER<16>	F-27
MER<15>	F-27
MER<14>	F-27
MER<13:12>	F-27

MER<11:8>	F-27
MER<7:0>	F-28
Memory Size Register (MSR)	F-28
MSR<31:27>	F-28
MSR<26>INTREN	F-28
MSR<25>MS10EN	F-28
MSR<24:21>PFORCE	F-28
MSR<20>MABEN	F-28
MSR<19:17>LASTBANK	F-29
MSR<16:14>	F-29
MSR<13>	F-29
MSR<12:0>	F-29
Error Address Register (EAR)	F-29

Index

Figures

1-1	Personal DECstation function blocks	1-4
1-2	System Unit	1-6
1-3	Controls and indicators	1-7
1-4	External system unit connectors	1-9
1-5	Internal base system module connectors	1-10
1-6	Clear-NVR jumper	1-11
1-7	System Boot ROM Switches	1-12
2-1	Alternate terminal	2-3
3-1	Installing a Console Terminal	3-2
5-1	Flash Sequence of the Diagnostic LED Indicator	5-12
5-2	External System Unit Connectors	5-15
9-1	Troubleshooting When the Monitor Has No Display	9-2
9-2	Troubleshooting a Missing SCSI Device	9-4
9-3	Troubleshooting a Missing TURBOchannel Option	9-6
9-4	Troubleshooting Missing Memory SIMMs	9-8
9-5	Troubleshooting a Missing SERIAL.bus Device	9-10
10-1	Using the antistatic kit	10-2
10-2	System unit cover	10-3
10-3	Serial line adapter	10-5
10-4	SCSI controller terminator	10-6
10-5	SCSI chain terminator	10-7
10-6	ThickWire Ethernet loopback connector	10-8

10-7	Serial line loopback connector	10-9
11-1	Internal drives	11-5
11-2	Floppy disk drive	11-8
11-3	Floppy disk drive	11-9
11-4	Floppy port cover	11-11
11-5	Removing and installing an internally mounted hard disk drive	11-13
11-6	RZ23L half-height hard disk drive	11-15
11-7	RZ23L SCSI ID jumper settings	11-16
11-8	RZ24 hard disk drive	11-17
11-9	RZ24 drive SCSI ID jumper settings	11-18
11-10	RZ24L hard disk drive	11-19
11-11	RZ25 hard disk drive	11-20
11-12	RZ25 drive front SCSI ID jumper settings	11-21
11-13	RZ25 drive side SCSI ID jumper settings	11-22
11-14	RZ26 hard disk drive front and back view	11-23
11-15	RZ26 hard disk drive	11-24
11-16	TLZ06 cassette tape drive	11-25
11-17	RX26 floppy disk drive	11-26
12-1	R3000 CPU module	12-3
12-2	R4000 CPU module	12-3
12-3	Removing and installing a CPU module	12-6
12-4	Identifying 2-megabyte memory modules	12-9
12-5	Identifying 8-megabyte memory modules	12-10
12-6	Memory module slot numbers	12-11
12-7	Removing a memory module	12-12
12-8	Installing a memory module	12-14
12-9	Removing and installing a TURBOchannel option module ...	12-16
12-10	Option module ESAR chip	12-18
12-11	Base system ESAR chip	12-20
12-12	Base system module	12-22
12-13	Removing and installing the base system module	12-24
12-14	Power supply	12-28
13-1	Keyboards	13-2
13-2	VSXXX-BB Mouse	13-3
13-3	VSXXX-GB Mouse	13-3
13-4	Removing and installing a keyboard and mouse	13-5

Tables

1	Conventions Used in This Guide	xxiv
3-1	Alternate Terminal Settings	3-3
5-1	Diagnostic Tests for Base System	5-6
5-2	Identification Codes for Selected TURBOchannel Options ...	5-6
5-3	Alternate Terminal Settings	5-7
5-4	Interpreting Diagnostic LED Indicator at Power-Up	5-13
5-5	Interpreting Diagnostic LED Indicator During 2nd 30 Seconds Following First Video (Power-Up Self-Test Period) ..	5-13
5-6	Interpreting Diagnostic LED Indicator During First 30 Seconds After Power-Up (Firmware Initialization Period) ...	5-14
6-1	Console Commands	6-3
6-2	Slot Numbers in Commands and Messages	6-11
6-3	Environment Variables	6-13
6-4	SCSI Controller Error Codes and Descriptions	6-15
6-5	Diagnostic Tests for Base System	6-16
6-6	Identification Codes for Selected TURBOchannel Options ...	6-16
6-7	Console Command Error Messages	6-17
7-1	Event Types	7-5
11-1	SCSI Cable Lengths	11-3
11-2	Internal Drive ID Settings	11-14
12-1	CPU Modules	12-2
12-2	Identifying Memory Modules	12-8
12-3	RAM address ranges	12-9
A-1	System Unit Description	A-2
A-2	System Unit Operating Conditions	A-2
A-3	System Unit Nonoperating Conditions	A-3
A-4	Acoustics—Preliminary Declared Values per ISO 9296 and ISO 7779	A-3
A-5	VRC16 Monitor Description	A-4
A-6	VRC16 Monitor Operating Conditions	A-5
A-7	VRC16 Monitor Nonoperating Conditions	A-5
A-8	VRM17 Monitor Description	A-6
A-9	VRM17 Monitor Operating Conditions	A-8
A-10	VRM17 Monitor Nonoperating Conditions	A-8
A-11	LK501-AA Keyboard Description	A-9
A-12	LK501-AA Keyboard Operating Conditions	A-10
A-13	LK501-AA Keyboard Nonoperating Conditions	A-10
A-14	VSXXX-BB Mouse Description	A-11

A-15	VSXXX-BB Mouse Operating Conditions	A-12
A-16	VSXXX-BB Mouse Nonoperating Conditions	A-12
A-17	RZ23L SCSI Hard Disk Drive Description	A-13
A-18	RZ23L SCSI Hard Disk Drive Operating Conditions	A-14
A-19	RZ23L SCSI Hard Disk Drive Nonoperating Conditions	A-14
A-20	RZ24L SCSI Hard Disk Drive Description	A-15
A-21	RZ24L SCSI Hard Disk Drive Operating Conditions	A-16
A-22	RZ24L SCSI Hard Disk Drive Nonoperating Conditions	A-16
A-23	RZ25 SCSI Hard Disk Drive Description	A-17
A-24	RZ25 SCSI Hard Disk Drive Operating Conditions	A-18
A-25	RZ25 SCSI Hard Disk Drive Nonoperating Conditions	A-18
A-26	RZ26 SCSI Hard Disk Drive Description	A-19
A-27	RZ26 SCSI Hard Disk Drive Operating Conditions	A-20
A-28	RZ26 SCSI Hard Disk Drive Nonoperating Conditions	A-20
A-29	TLZ06 Cassette Tape Drive Description	A-21
A-30	TLZ06 Cassettes	A-21
A-31	TLZ06 Cassette Tape Drive Operating Conditions	A-21
A-32	TLZ06 Cassette Tape Drive Nonoperating Conditions	A-21
A-33	RX26 Diskette Drive Description	A-22
A-34	RX26 Diskette Disk Drive Specifications (formatted)	A-22
A-35	RX26 Diskette Drive Operating Conditions	A-23
A-36	RX26 Diskette Drive Nonoperating Conditions	A-23
B-1	Basic System Components	B-2
B-2	SCSI and Floppy Disk Hardware	B-3
B-3	Monitors	B-4
B-4	Input Devices	B-4
B-5	Cables and Cords	B-5
B-6	Miscellaneous Hardware	B-5
B-7	Connectors, Adapters, and Terminators	B-6
B-8	Hardware Documentation	B-7
B-9	Software Documentation	B-8
B-10	Miscellaneous TURBOchannel options	B-8
B-11	TURBOchannel Frame buffer options	B-9
B-12	TURBOchannel Graphic accelerator options	B-10
C-1	SCSI Connector Pin Assignments	C-2
C-2	ThickWire Ethernet Connector Pin Assignments	C-3
C-3	EIA RS232 Communications Connectors Pin Assignments	C-4
C-4	Power Supply Pin Assignments	C-6
C-5	ISDN Connector	C-7

C-6	SERIAL.bus Pin Assignments	C-8
C-7	Sound Connector	C-9
C-8	Summary of Loopback Connectors	C-10
C-9	Ethernet Loopback Connector Pin Assignments, Part #12-22196-02	C-10
C-10	Modem Loopback Connector Pin Assignments, Part #12-15336-13, Option #H3200	C-11
D-1	Console Command Functions	D-6
D-2	Standard Environment Variables for the Personal DECstation 5000	D-32
D-3	Console Command Error Messages	D-38
E-1	Base system module tests and utilities	E-2
E-2	Desktop Bus Device Capabilities Test Error Codes and Descriptions	E-6
E-3	Cache Data Test Error Codes	E-9
E-4	Cache Fill Test Error Descriptions	E-10
E-5	Cache Isolate Test Error Codes	E-12
E-6	Cache Reload Test Error Descriptions	E-13
E-7	Cache Segment Test Error Codes and Descriptions	E-15
E-8	Frame Buffer Test Error Codes and Descriptions	E-17
E-9	EEPROM Signature Data Check Test Error Descriptions ...	E-20
E-10	Free-running Counter Interval Test Error Descriptions	E-21
E-11	Floppy DOR Register Test Error Codes and Descriptions	E-22
E-12	Exercise Floppy Disk Controller Test Error Codes and Descriptions	E-23
E-13	Floppy Media Read Test Error Descriptions	E-24
E-14	Reset Floppy Disk Interface Test Error Descriptions	E-25
E-15	Write/Read Floppy Media Test Error Descriptions	E-27
E-16	FPU Test Error Codes	E-28
E-17	ISDN Baseline Test Error Descriptions	E-29
E-18	Partial Write Test Error Codes	E-34
E-19	Collision Test Error Codes and Descriptions	E-35
E-20	Common Diagnostic Utility Error Codes and Descriptions ...	E-36
E-21	CRC Test Error Codes and Descriptions	E-37
E-22	Ethernet-DMA Registers Test Error Codes and Descriptions	E-38
E-23	Ethernet-DMA Registers Test Error Codes and Descriptions	E-40
E-24	ESAR Test Error Codes and Descriptions	E-41
E-25	External Loopback Test Error Codes and Descriptions	E-42

E-26	IRQ Test Error Codes and Descriptions	E-43
E-27	Internal Loopback Test Error Codes and Descriptions	E-44
E-28	Multicast Test Error Codes and Descriptions	E-45
E-29	Promiscuous Mode Test Error Codes and Descriptions	E-47
E-30	Registers Test Error Codes and Descriptions	E-48
E-31	LANCE-setup utilities error codes and Descriptions	E-49
E-32	RTC Period Test Error Codes	E-50
E-33	Real-Time Clock Register Test Error Codess	E-51
E-34	Real-Time Test Error Codes	E-52
E-35	SCC DMA Test Error Codes	E-55
E-36	SCC I/O Test Error Codes and Descriptions	E-57
E-37	Pin Pairs Tested by Individual Loopback Connectors	E-58
E-38	SCC Pins Test Error Codes and Descriptions	E-59
E-39	SCC Transmit and Receive Test Error Codes and Descriptions	E-61
E-40	SCSI Controller Error Codes and Descriptions	E-62
E-41	SCSI Send Diagnostics Test Error Error Codes and Descriptions	E-63
E-42	SCSI Target Test Error Codes and Descriptions	E-65
E-43	TLB Registers Test Error Descriptions	E-67
E-44	Power-Up LED Codes	E-68
F-1	R3000A CPU Registers	F-2
F-2	R4000 CPU Registers	F-3
F-3	Exception Codes R3000	F-5
F-4	Exception Codes R4000	F-6
F-5	Diagnostic Status Fields	F-13
F-6	Config Registers	F-15
F-7	System Registers	F-16
F-8	System Support Register 0xBC040100	F-17
F-9	System Interrupt Register 0xBC040110	F-21
F-10	System Interrupt Mask Register 0xBC040120	F-26
F-11	Memory Error Register 0xAC400000	F-27
F-12	Memory Size Register 0xAC800000	F-28

About This Guide

This guide explains how to perform maintenance operations on the Personal DECstation/DECsystem 5000 workstation and server. The following topics are discussed:

- The parts that make up the basic Personal DECstation /DECsystem 5000 workstation and server
- The meaning of icons that appear on the system unit and cables
- The function of each workstation component
- The function of the connectors, controls, switches, indicator lights, and LED display on the base system unit
- Site and voltage requirements
- How to set up the workstation and adjust workstation hardware
- What options are available to improve workstation performance
- How to install and remove optional internal workstation components
- How to connect a workstation to an Ethernet network
- How to connect external communications options
- How to install a terminal as a separate system console
- How to connect external storage devices to a workstation
- How to power up, test, and shut down a workstation

- How to read the configuration displays and interpret the power-up and system self-test results
- How to diagnose and solve basic hardware problems
- How to use console commands
- Specifications and part numbers for workstation components
- Pin assignments for workstation cables and connectors

Table 1. Conventions Used in This Guide

Convention	Use
Monospace type	Anything that appears on the monitor is set in monospace type, like this.
Boldface type	Anything you are asked to type is set in boldface type, like this.
Caution note	Provides information that protects the workstation from being damaged.
Warning note	Provides information that protects you from being harmed.

Part I

System Overview

Base System

This chapter provides an overview of the following Personal DECstation/DECsystem 5000 hardware:

- Basic system hardware
- System hardware configurations

System Hardware Configurations

The Personal DECstation/DECsystem contains a rich set of maintainability tools and features. The troubleshooting section of the manual describes, in some detail, four techniques to detect and isolate problems.

The Personal DECstation/DECsystem is a powerful yet flexible machine. Although it is the entry-level member in the Digital line of MIPS Technologies Inc, (MIPS) RISC workstations, it supports all the TURBOchannel options and peripherals Digital has developed for its high-end workstations. The system also works with many industry-standard commodity peripherals, such as monitors and disk drives, sold by third party vendors.

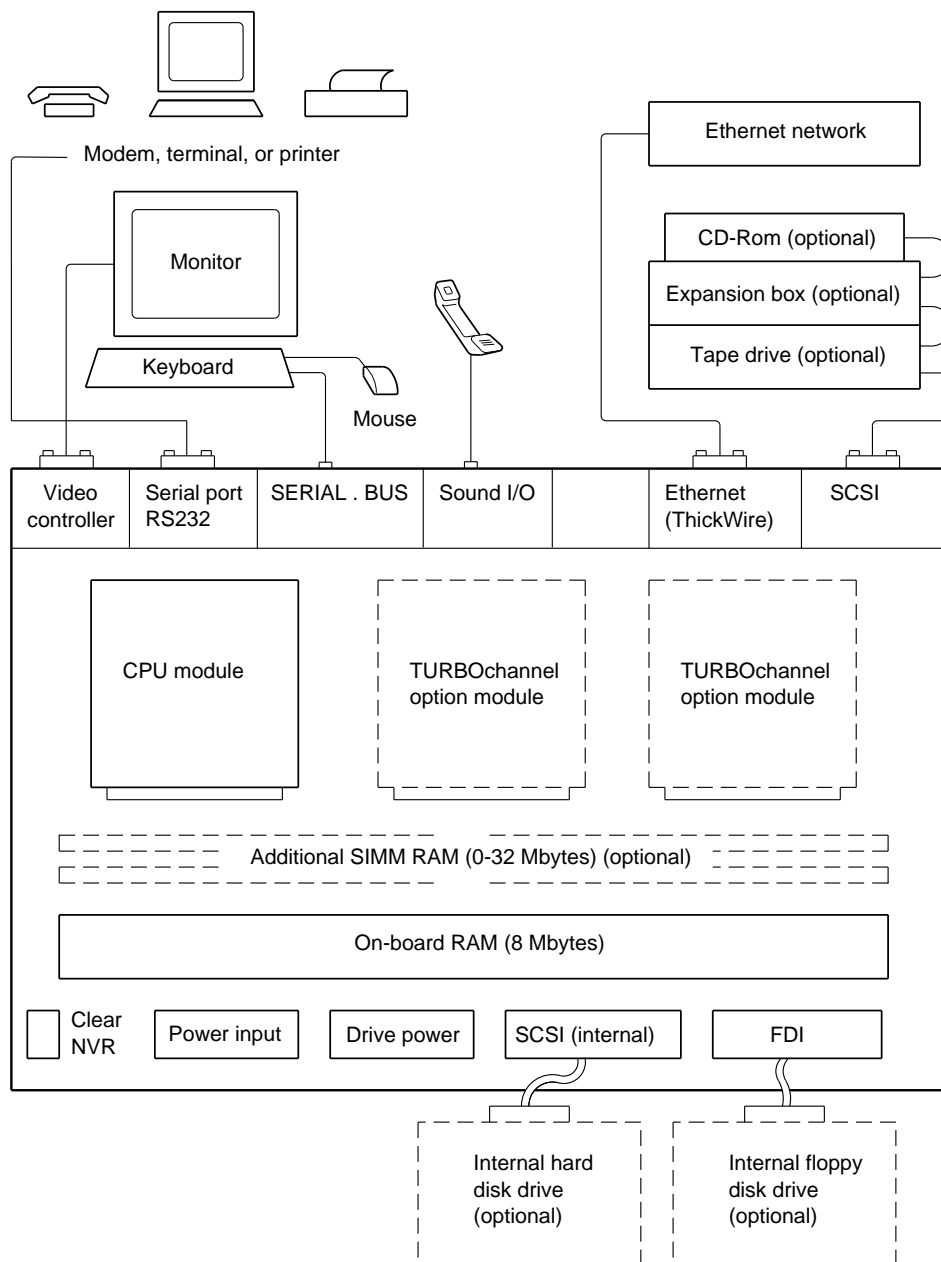
The Personal DECstation/DECsystem can be configured in many ways. This can range from a simple standalone server to a multiscreen workstation with multiple communication connections and many external storage devices.

The Personal DECstation is a MIPS, R3000-based, reduced instruction set computer (RISC) desktop system designed to support the ULTRIX operating systems.

The Personal DECstation/DECsystem Model 50 is a MIPS Technologies Inc, R4000-based, reduced instruction set computer (RISC) desktop system designed to support the ULTRIX operating system.

- The Personal DECstation/DECsystem can be configured as either a workstation or a server.
 - When it is configured as a workstation, the user input and output devices are a graphics monitor, a keyboard, and a mouse.
 - When it is configured as a server, the user input and output devices are a terminal with keyboard connected to the serial communications port.
- The base system can be connected to the following devices:
 - One color or gray scale monitor
 - One keyboard and mouse
 - Up to seven SCSI storage devices (one mounted internally)

- One internally mounted floppy disk drive
- One Ethernet network
- A single modem, terminal, printer, or other serial device
- Additional monitors, storage devices, network connections, and serial devices can be supported by adding TURBOchannel option modules.



WSMXM001

Figure 1-1. Personal DECstation function blocks

System Unit

The system unit is the assembly of the modular components of the workstation, including the system unit chassis and cover, the modules, and the power supply. The modular components are called field replaceable units (FRUs). The components and features of the system unit are listed here and shown in Figure 1-2.

- Base system module, which includes
 - One CPU module connector
 - Eight megabytes of 32-bit parity random access memory (RAM), resident on the system module
 - Four connectors for up to 32 megabytes of additional RAM
 - One on-board 1024 x 768, 8-plane, 72-hertz color frame buffer video controller
 - One SERIAL.bus interface for keyboard and mouse
 - One floppy disk interface (FDI) controller for an internally mounted 3.5-inch floppy disk drive (such as the RX26)
 - One small computer system interface (SCSI) controller for up to seven drives
 - One synchronous/asynchronous RS232 serial communications port with full modem control
 - One ThickWire Ethernet controller
 - Two connectors for TURBOchannel option modules
 - Diagnostic ROM
- R4000 has one removable CPU module with a floating-point unit (FPU), cache memories, and a ROM that supercedes the base system ROM.
- R3000 has one removable CPU module with a floating-point unit (FPU), cache memories.
- Space for two internal drives

- One power supply with cooling fan
- One metal chassis
- One removable cover

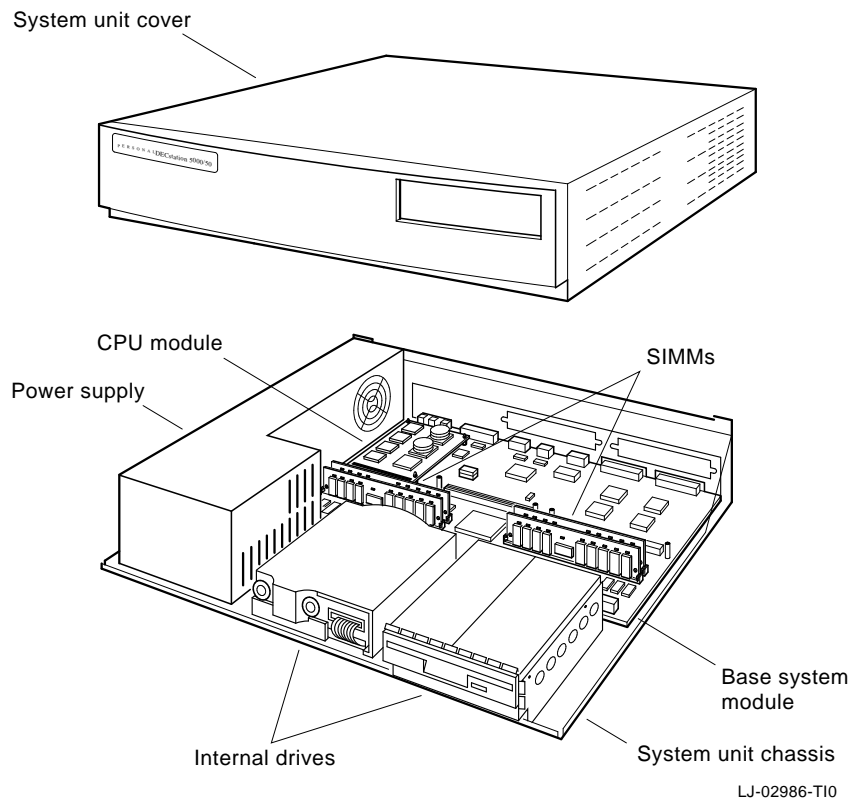


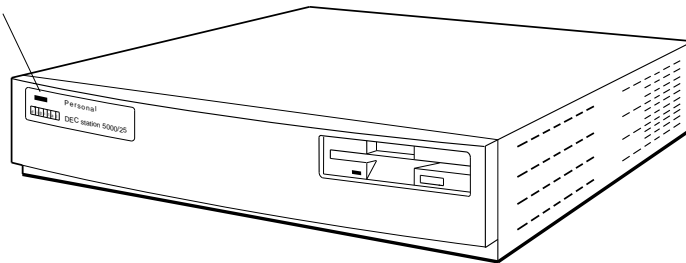
Figure 1-2. System Unit

Controls and Indicators

The system unit controls and indicators are listed here and shown in Figure 1-3.

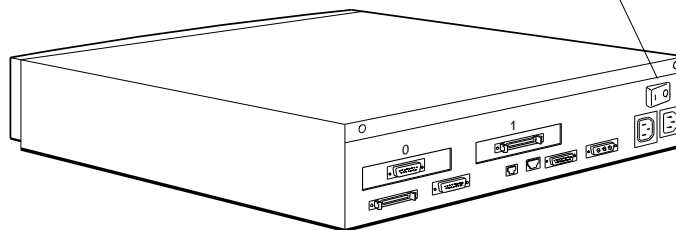
- The on/off switch controls the power to the system unit. Press down the 1 side of the switch to turn the power on. Press down the 0 side of the switch to turn the power off.
- The status indicator LED glows green when the power supply is operating properly and displays blinking error codes to identify faulty components. When the cover is installed, the status indicator LED illuminates the medallion on the front of the system unit.

Status indicator
LED (green)



Front view

On/off switch



Back view

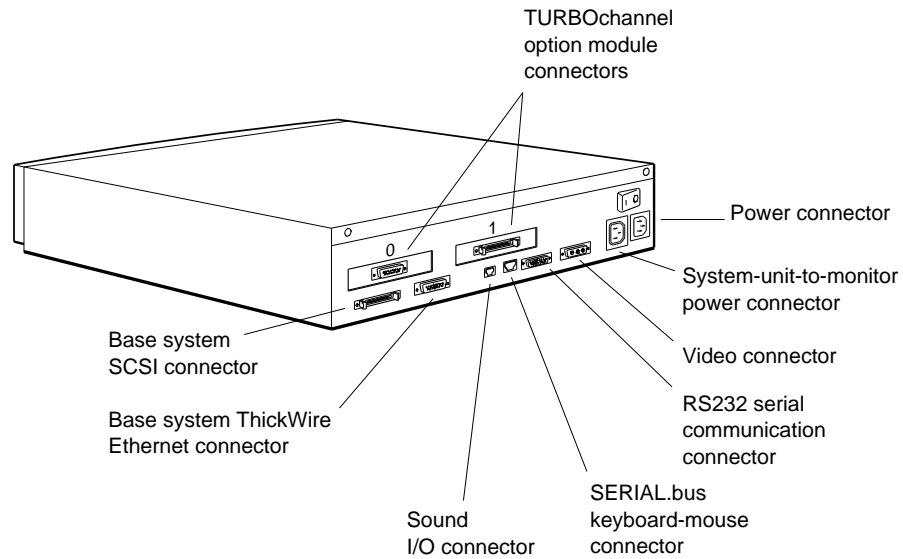
WSMXM004

Figure 1-3. Controls and indicators

External System Unit Connectors

The external system unit connectors on the back panel connect the workstation to external devices. The external system unit connectors are listed here and shown in Figure 1-4.

- The power connector connects the system unit power supply and the system unit power cord.
- The system-unit-to-monitor connector allows the monitor to receive its electric power through the system unit rather than directly from a wall outlet.
- The video connector links the base system video controller to a color or gray scale monitor.
- The SERIAL.bus connector links the system unit to the keyboard-mouse cable.
- The base system ThickWire Ethernet connector links the base system Ethernet controller to an Ethernet network.
- The base system external SCSI connector connects the base system SCSI controller to a chain of external drives.
- The RS232 serial communications connector links the base system serial line controller to an external device, such as a printer, modem, or console terminal.
- The two option module connectors, labeled 0 and 1, connect TURBOchannel option modules to external hardware. The number printed above each option module position is the logical slot number by which an option module mounted in that position is addressed in test commands and identified in error messages.



WSMXM005

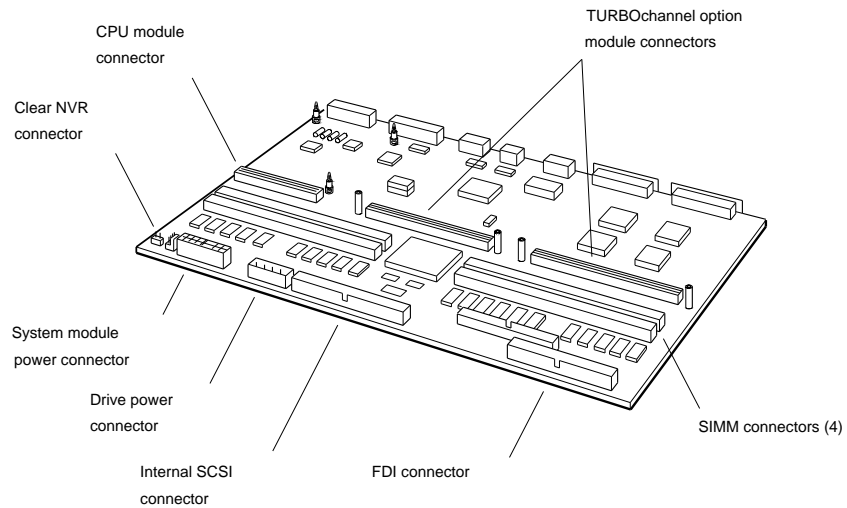
Figure 1-4. External system unit connectors

Internal Base System Module Connectors

The internal base system module connectors are the means by which the system's modular components, both standard and optional, are connected to the base system module and to each other. The internal base system module connectors are listed here and shown in Figure 1-5.

- The CPU module connector connects the CPU module to the base system module.
- Two pairs of SIMM connectors provide the means to install up to 32 megabytes of additional RAM.
- Two internal TURBOchannel option module connectors connect TURBOchannel option modules to the base system module. The connector closest to the power supply is referred to as logical slot number 1 in test commands and error messages. The connector furthest away from the power supply is logical slot 0.
- The system module power connector receives DC power from the power supply for all components in the system unit enclosure.

- The drive power connector supplies DC power to the two internally mounted drives.

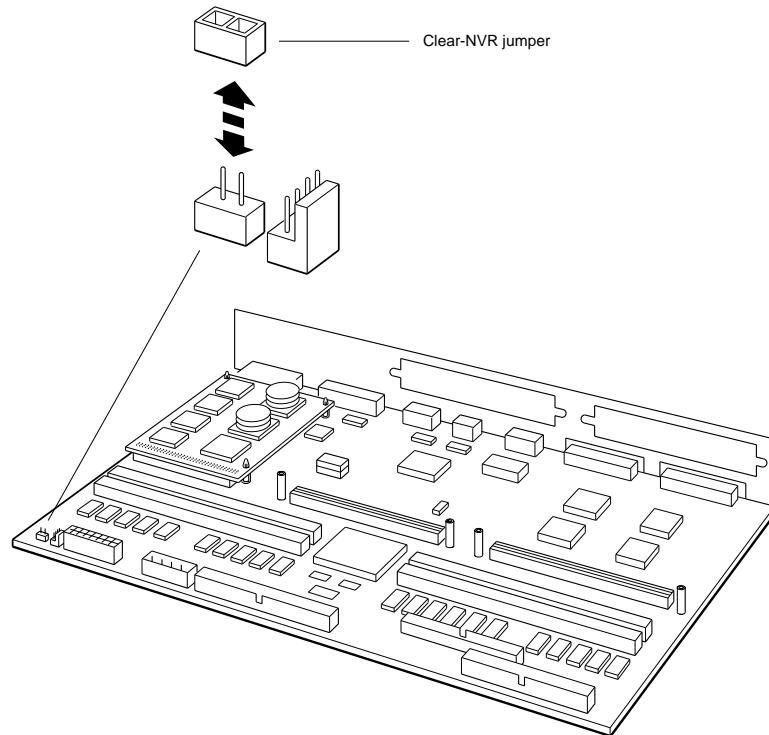


WSMXM006

Figure 1-5. Internal base system module connectors

Clear-NVR Jumper

The clear-NVR jumper (12-14314-00) erases any existing environment variable settings by shorting together the two points on the connector and power cycling the machine.



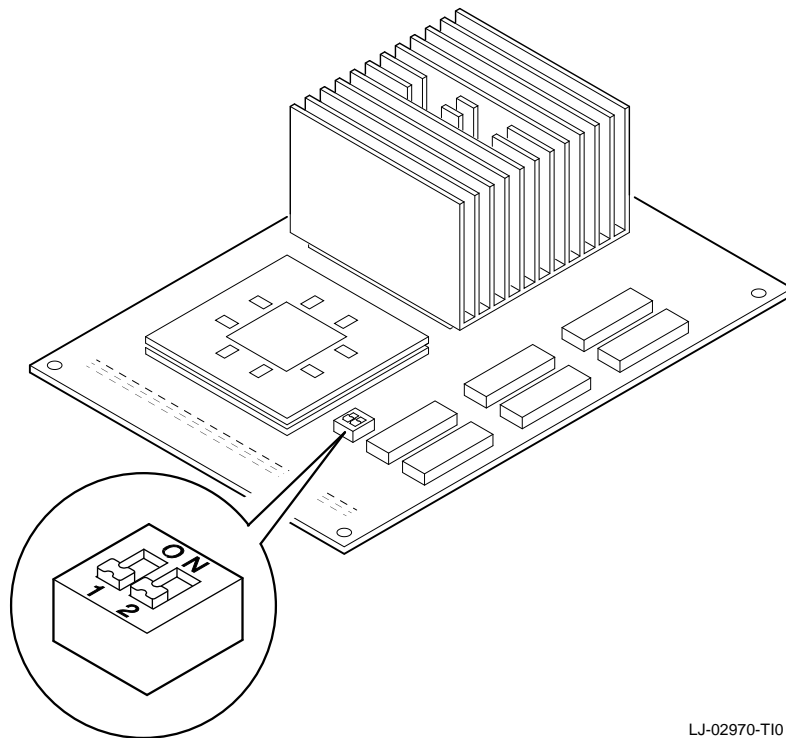
WSMXM007

Figure 1-6. Clear-NVR jumper

System Boot ROM

The Boot ROM on the system module does not need to be replaced because there is a Boot ROM on the R4000 CPU. Two switches on the module control access to the Boot ROM. Both switches must be off. Refer to Figure 1-7 for location of switches.

Note: Note the orientation of the switches. Switches pointing to the ON position are on, switches pointing towards the 1, 2 are off.



LJ-02970-T10

Figure 1-7. System Boot ROM Switches

For Further Information

For troubleshooting information, see Chapters 4, 5, 6, 7, and 9 of this guide.

For detailed information about system unit FRUs, see Chapters 10 and 11 of this guide.

For information about Personal DECstation/DECsystem hardware options and peripherals, see Chapter 2 of this guide.

Hardware Options and Peripherals

This chapter provides an overview of the following options and peripherals supported on the Personal DECstation/DECsystem hardware:

- One CPU module (R4000 or R3000 based)
- Two or four memory modules for up to 32 megabytes of additional RAM
- One or two TURBOchannel option modules
- Up to seven SCSI drives per SCSI controller
- One internally mounted floppy disk drive (FDI)
- Graphics monitors and terminals
- A keyboard and one mouse
- RS-232 serial devices

Figures 1-4 and 1-5, show the external and internal connectors that support the hardware options and peripherals.

CPU Module

The Personal DECstation supports the following CPU modules:

- R3000 CPU modules
 - 25-megahertz CPU with 128-kilobyte cache (KN02-CC)
 - 33-megahertz CPU with 128-kilobyte cache (KN02-DA)

- R4000 CPU module
 - 50-megahertz CPU with 1-Megabyte secondary cache (KN04-AA)

See “Removing and Replacing a CPU Module” in Chapter 12 for detailed information about CPU modules.

Memory Modules

The base system has eight megabytes of 32-bit parity RAM resident on the base system module. In addition, the base system module has two pairs of connectors for additional memory. Each connector accepts one Standard In-line Memory Module (SIMM). The following SIMM types are supported:

- 2-megabyte SIMM (MS01-AA or MS01L-AB)
- 8-megabyte SIMM (MS01-CA)

Do not mix 2-megabyte and 8-megabyte SIMMs in a single system. The SIMMs must be installed in pairs starting with the paired memory connectors numbered 0, nearest the back of the system. See “Removing and Installing Memory Modules” in Chapter 12 for information about memory modules.

Monitors and Terminals

The on-board 1024 x 768, 72-hertz, 8-plane color frame buffer video controller currently supports the following monitors:

- VRC16 sixteen-inch color monitor
- VRM17 seventeen-inch gray scale monitor
- A third-party 1024 x 768, 72-Hz, color or gray scale monitor

If optional TURBOchannel graphics modules are installed, the monitors supported depend on the TURBOchannel graphics modules installed. Refer to the *TURBOchannel Maintenance Guide* for a list of monitors that each TURBOchannel graphics module supports.

You can connect an alternate console terminal to the serial communications connector on the back of the system unit. The monitor can be any RS232 monitor of type VT52 or

greater. Figure 2-1 shows a monitor and an alternate terminal connected to the system unit.

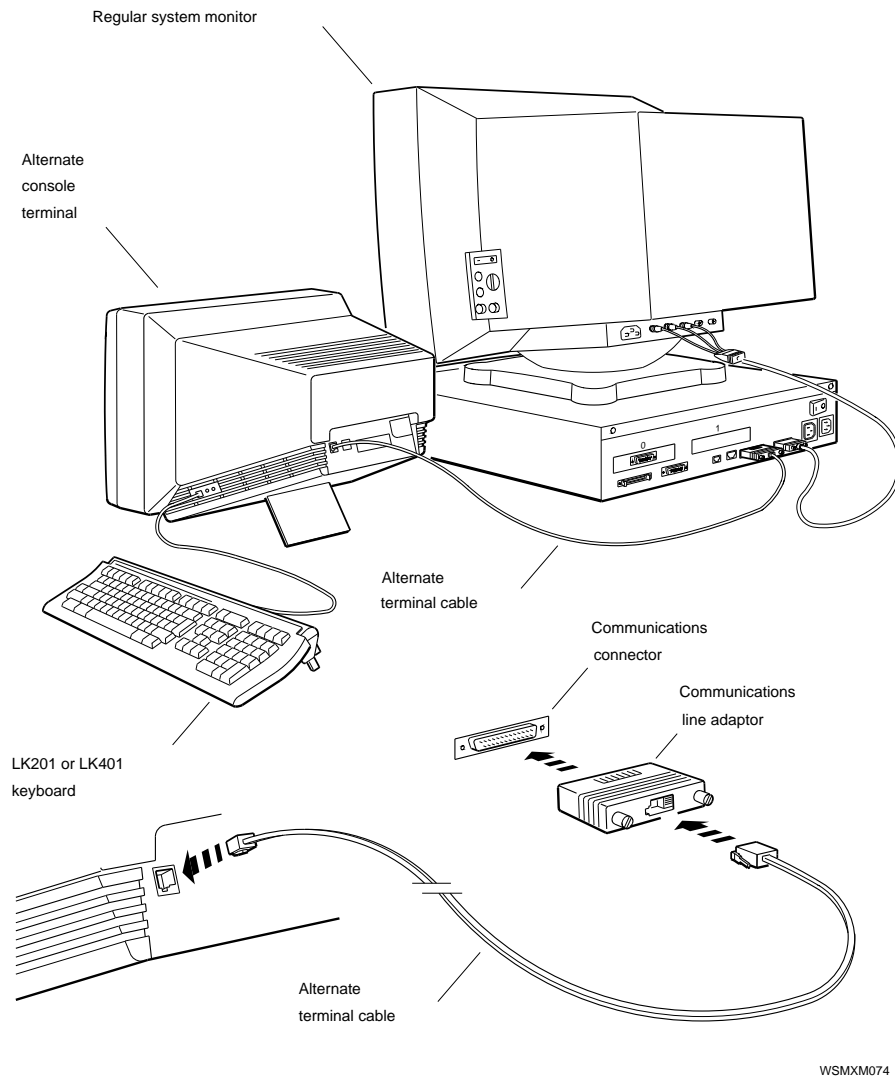


Figure 2-1. Alternate terminal

Keyboard and Mouse

The system receives input from the user through an LK501 keyboard and a VSXXX-GB mouse. The keyboard and mouse connect to the SERIAL.bus keyboard-mouse connector on the

back of the system unit through the keyboard-mouse cable. See Chapter 13 for information about the keyboard and mouse.

SCSI Drives

The base system module comes with a SCSI controller. In addition, one or two TURBOchannel SCSI controller option modules can be added to the system. Each SCSI controller can support up to seven drives. One of the drives connected to the base system SCSI controller can be mounted inside the system unit enclosure. See Chapter 11 for more information about the SCSI drives.

TURBOchannel Option Modules

The TURBOchannel connectors on the base system module can support two TURBOchannel option modules according to the following rules:

- Any SCSI controller, Ethernet controller, or serial interface TURBOchannel option modules operate in addition to the equivalent functions on the base system module. See “Removing and Installing TURBOchannel Option Modules” in Chapter 12 for information about removing and installing TURBOchannel option modules.
- A TURBOchannel option module that occupies three spaces, such as a PMAG-FA or PMAGB-FA module, must be installed in a TURBOchannel Extender box and connected to a TURBOchannel Extender option module installed in one of the TURBOchannel option slots of the Personal DECstation/DECsystem.
- When one or more TURBOchannel option graphics modules are installed, the following rules for multiscreen use apply:

- Two frame buffer modules can be used in multiscreen mode only if they are frame buffers of the same type.
- A 1024 x 768, 8-plane, 72-hertz color frame buffer module can be used in multiscreen mode with the base system video controller, which is of the same type. (Use the HX PMAGB-BE module.)
- The TURBOchannel option graphics module installed in the lowest-numbered slot, not the base system video controller, will be used as the output console device.

For Further Information

For troubleshooting information, see Chapters 4, 5, 6, 7, and 9 of this guide.

For detailed information about the FRUs mentioned in this chapter, refer to Chapters 10 and 11 of this guide.

For further information about field service of TURBOchannel option modules and connected drives and monitors, refer to the *TURBOchannel Maintenance Guide* and the *TURBOchannel Extender Maintenance Guide*.

A Server Configuration

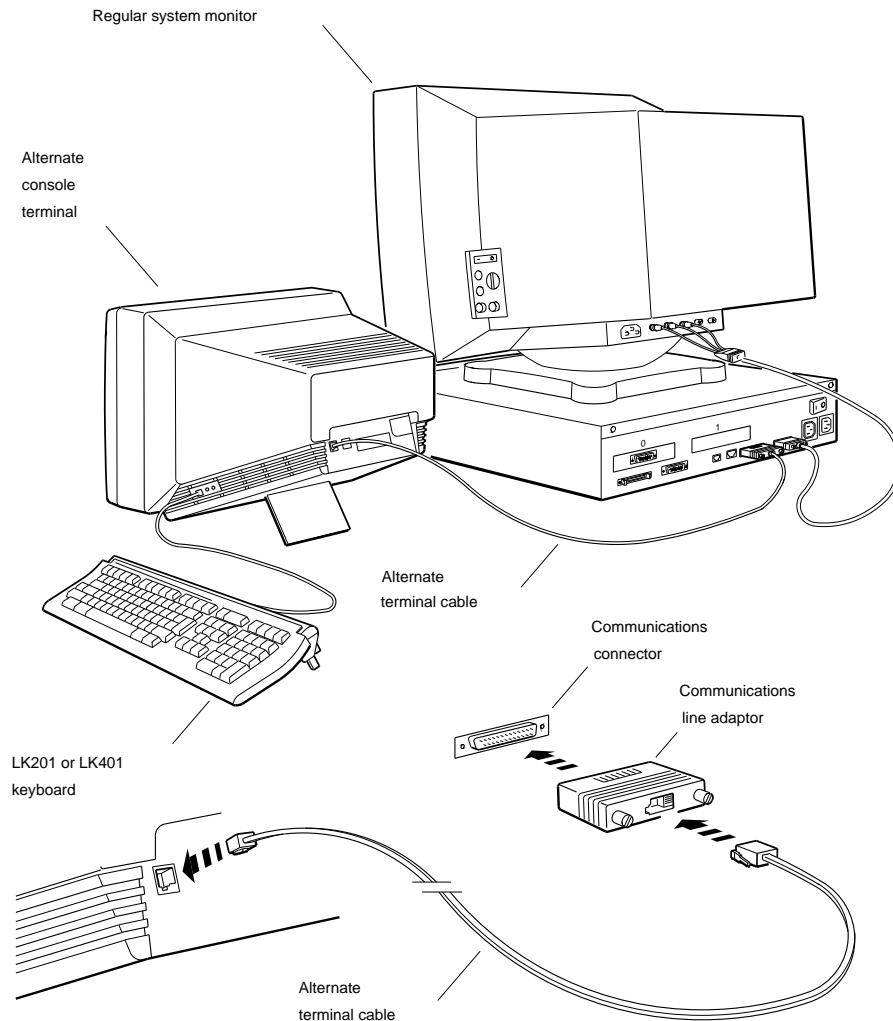
This chapter discusses the following topics:

- Connecting a Console Terminal
- Halting a Server

Introduction

The Personal DECstation/DECsystem has been designed to be configured as either a workstation or a server. (A server is used to provide other client systems in a network with distributed services such as filing, computation and printing.)

When performing distributed network server tasks, the Personal DECstation/DECsystem has no need for the expensive graphics hardware usually associated with workstations, so it is configured without a graphics monitor. However, to maintain a console window, connect the Personal DECstation/DECsystem to the input and output of a VT320-type terminal attached to the RS232 port in the rear of the system unit, as shown in Figure 3-1.



WSMXM074

Figure 3-1. Installing a Console Terminal

Connecting a Console Terminal

Connect your terminal to the 25-pin RS232 connector, and set your terminal as described in Table 3-1. You may notice that these settings are the standard default settings for Digital terminals. Some terminals may use the modular MMJ

telephone style connectors. If this is the case, an adapter (Part #H8571-A) can be used to make the connection.

Note: *Customers who have multiple servers may not have a terminal plugged into each unit. In this type of configuration, the system manager generally accesses the server via the Ethernet.*

Table 3-1. Alternate Terminal Settings

Setting	Value
Baud rate	9600
Bits	8
Parity	None
Stop bits	1
Flow Control	Xon/Xoff
Mode	Full duplex

Once you have connected the terminal to the RS232 port, you can now use the messages displayed on the terminal to perform system operations.

Halting a server

A server generally does not need to be halted during normal operations. If a server does need to be halted, it is usually shut down from the operating system with the *shutdown* command. However, applications developers and other advanced users may at times wish to abruptly halt the server to examine the contents of memory during code development and debug.

The Personal DECstation/DECsystem may be abruptly halted as a server by the same technique that is used when it is abruptly halted as a workstation: namely, plug in an LK501 key board and type **Ctrl-Alt-Return**. We recommend that developers who do this type of work also set the console environmental variable to **s**, as explained in the next section.

Note: *The Personal DECstation/DECsystem does not have an explicit Halt switch.*

Default Settings for Using the Console Terminal

An advanced user may wish to alternate back and forth between use of a graphics monitor and a console terminal. When using a console terminal, defeat the default power-up console assignment by issuing the following console command:

```
>> setenv console s
```

The console can be switched back to a graphics monitor with the command:

```
>> setenv console *
```

Operational background information

The system has the ability to display console and diagnostic messages, either on the graphics monitor, or on a VTxxx character-based terminal connected to the RS232 port located on the rear of the unit.

At powerup, the system looks for a keyboard plugged into the SERIAL.bus. If one is found, the system will route all console messages to the graphics monitor. If, however, no keyboard is detected on the SERIAL.bus, all text will be sent directly to the RS232 port.

Therefore, when using the Personal DECstation/DECsystem as a server and there is an LK501 keyboard plugged in when the system is powered on, the system will attempt to display console output on a non-existent graphics monitor. To prevent this from happening, establish the terminal as the default console device by issuing the **setenv console s** command, as described in the previous section.

Part II

Troubleshooting

Troubleshooting Overview

This chapter discusses the following topics:

- Troubleshooting overview
- The troubleshooting process
- Four troubleshooting techniques
- Procedural flow charts
- General troubleshooting tips
- Obtaining a Crash Dump

Troubleshooting Overview

Just as the complexity of the configurations can vary, so can the complexity of the problems encountered. Some problems are fairly straightforward to diagnose. Others problems, like intermittent hangs and lost network traffic, have much more subtle causes. Some problems may involve interactions between the operating system, the application software, and the hardware.

Four Troubleshooting Techniques

The next four chapters discuss the tools available and provide the necessary background information to successfully troubleshoot a Personal DECstation.

When isolating Personal DECstation/DECsystem problems usually start with the power-up self-test (PST). If greater detection and isolation capabilities are needed to resolve the problem, the console and diagnostic tools can then be used. Finally, for problems with more subtle origins, use the error logger and system exerciser.

The approach to troubleshooting presented in this manual starts with a fairly general test tool, and then moves on to tools that do a progressively more specific job of identifying your problems.

By resolving any difficulties encountered when using the general tools, subtle problems are not masked or compounded by other failures.

- Chapter 5 “The Power-Up Self-Test”
Discusses how to use ROM-based power-up self-test diagnostics, and how to connect an alternate terminal. It also describes how to interpret the flashes the system LED displays during power-up.
- Chapter 6 “ROM Based Console and Diagnostics”
Discusses how to run the diagnostics scripts and test, and interpret their error messages.
- Chapter 7 “ULTRIX Error Reports and System Exerciser”
Discusses how the ULTRIX error reports and system exercisers can be used to isolate intermittent problems.
- Chapter 8 “Obtaining a Crash Dump”
Discusses how to force a crash dump on a hung system. A dump provides a snapshot of memory just prior to the system crashing or hanging.

Reference Material

The appendices in this guide amplify the material presented in the troubleshooting chapters. The appendices listed here are particularly useful as troubleshooting aids.

- Appendix D
Contains a detailed description of each console command.
- Appendix E
Contains a detailed description of the diagnostics tests and the error messages they produce.
- Appendix F
Contains a description of the system registers and memory address assignments.

General Troubleshooting Tips

Below are a few tips and questions you should ask when troubleshooting.

- It is as important to know what works as it is to know what is broken.
- What malfunction does the user report?
- What malfunction does the field engineer observe?
- Has the system ever worked before?
- What changes occurred that could have caused the system to fail?
- Do you have good electrical contacts on all cables, connectors, sockets, and chips?
- Is there power, and is it getting to all parts of the system?
- Does the system see all the equipment you physically see?
- Is the system configured properly? (Check boot path, default console, halt action, etc.).
- If you have swapped the same board several times you probably have a software problem.
- What resources do you have to work with?
- Always read the table of contents and the index of your maintenance manual.
- Look for patterns in intermittent problems.

The Powerup Self-test

This chapter discusses the following topics:

- Troubleshooting Techniques
- Invoking the Powerup Self-Test
- Interpreting Video Error Messages
- Connecting and Using an Alternate Terminal
- Minimum Configuration Troubleshooting with the System LED
- Understanding the Powerup Self-test Sequence and the LED Display
- What to do when the Powerup LED Indicator is Inactive

Troubleshooting Techniques

The first step in isolating and detecting problems on a Personal DECstation/DECsystem is to interpret the results of the system's ROM-based power-up self-test. One of the three procedures described in this section can be used to identify the problem module. These techniques will be explained in detail in this chapter.

1. Troubleshooting with the graphics monitor

The primary method of troubleshooting a system is to interpret the console error messages which are displayed on the video screen during the power-up self-test. Identify the problem module and replace it. The power-up self-test should then be rerun to verify the system is operational.

2. Troubleshooting with the alternate video

Some failures can prevent the system from displaying error messages on the graphics monitor. In this case, connect a VTXXX terminal or equivalent to the RS232 port on the back of the unit. Use this alternative video display to troubleshoot the system.

3. Troubleshooting with the system LED

Customers who do not have access to a VTxxx-type terminal or equivalent can use the status LED as a troubleshooting aid. A single green LED, mounted to the front of the power supply, flashes several times during the system power-up. These flashes indicate how far the unit has progressed through the power-up sequence.

By interpreting these flashes one can determine which modules are operational and which modules are candidates for replacement. How to interpret the system LED is described in detail later in this chapter.

Invoking the Powerup Self-Test

The most straightforward way to invoke the power-up self-test diagnostics is to power-cycle the system. This is done by simply turning the power off and then turning it back on.

Note: *Power-cycling the machine will result in the erasure of the contents of memory.*

You can also type the commands **pst-q** or **pst-t** at the console prompt. **pst-t** will run a thorough test. **pst-q** will run a reduced version of the power-up self-test. This version tests all major sub-systems, but has been optimized to execute quickly.

When running the power-up self-test, it is important to either have a device connected to the Ethernet and SCSI ports, or to install the Ethernet loopback connector (Part #12-22196-02) and the SCSI terminator (Part #12-29635-01). If nothing is connected to these ports, the system may report false errors.

The Ethernet diagnostics used by **pst-t** actually send data out onto the network and then listen for a response. If none is received, an error is reported. The loopback connector routes this transmitted message back to the system, thereby enabling the Ethernet diagnostic to test the complete circuit path.

The basic design and architecture of the SCSI bus requires that a terminator be placed on the connector at the end of the bus. The SCSI I/O bus allows you to daisy chain up to seven devices together. These devices are most often disk drives that use the SCSI bus as a party line to communicate with the system.

Note: *This chapter focuses on how to use the power-up self-test diagnostics to isolate problem modules. There are additional diagnostics available. Their use is described in Appendix F, "Console Commands."*

Interpreting Video Error Messages

When the power-up self-test diagnostics encounter a failure, a formatted error message is displayed. The example below shows a breakdown of an error message which indicates that a problem was detected with the SCSI control portion of the base module.

The key pieces of information which lead to this conclusion are that the failure was detected on TURBOchannel slot 3, the base module. The problem was with the SCSI sub-system, and was specifically the controller. Finally, the diagnostic identified the failed module as the KN02-CA.

Example: Power-up Selftest Failure

Error Message Format:

?TFL slot_number/test_name (n:description) [module]

Example:

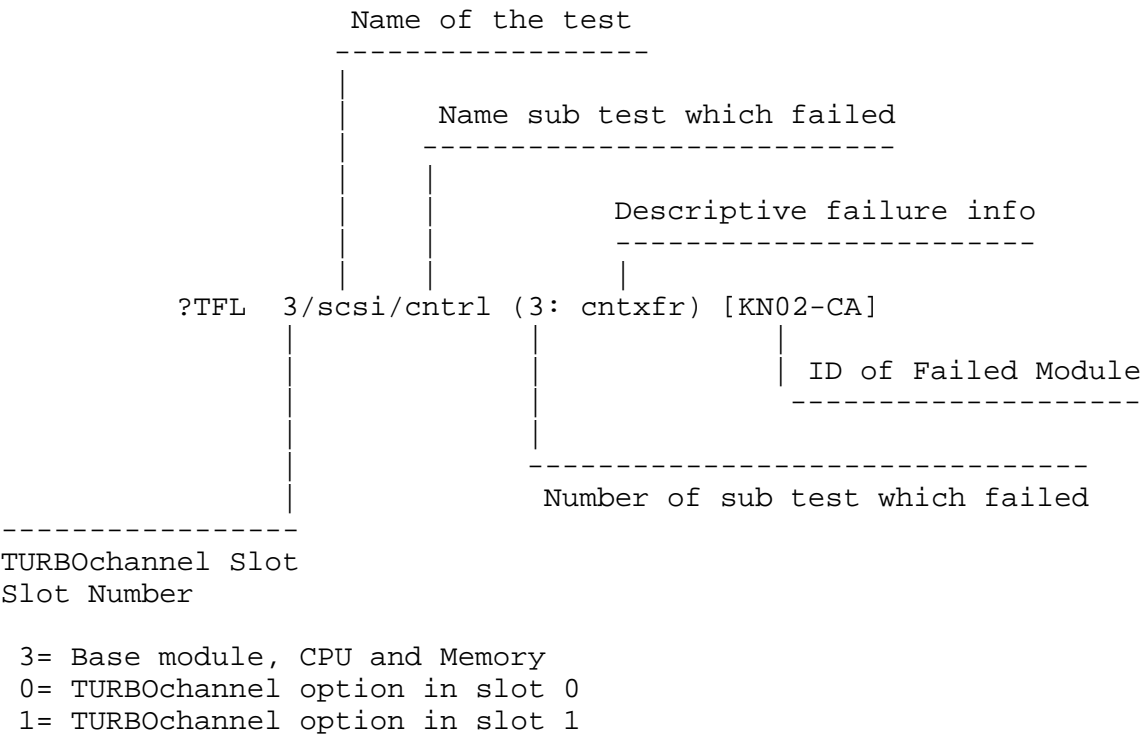


Table 5-1 and Table 5-2 contain reference material that is useful when interpreting error messages.

Table 5-1. Diagnostic Tests for Base System

Test	Sub-system	Problem module
ab	SERIAL.bus	Base module, keyboard, or mouse
cache	CPU	CPU daughter card
crt	Video frame buffer	Base module
epm	EEPROM	Base module
fdi	Floppy disk	Base module or floppy drive
fpu	CPU	CPU daughter card
isdn	Isdn and sound	Base module, ISDN peripherals
mem	Memory	Base module, memory SIMMs
ni	Ethernet	Base module, Enet card, Enet peripherals
rtc	Real time clock	Base module
scc	RS232	Base Module, RS232 peripherals
scsi	SCSI	Base Module, SCSI card, SCSI peripherals
scache	Secondary cache	CPU daughter card

Table 5-2. Identification Codes for Selected TURBOchannel Options

Module ID	Description
KN02-CA	Base module, R3000 CPU, memory
KN04-AA	Base module, R4000 CPU, memory
PMAG-AA	MX mono frame buffer
PMAG-BA	CX color frame buffer
PMAG-CA	PX 2D graphics accelerator
PMAG-DA	PXG 3D graphics accelerator
PMAG-FA	PXG turbo graphics accelerator
PMAGB-BA	HX smart frame buffer
PMAG-JA	TX true color frame buffer
PMAD-AA	NI thickwire Ethernet
PMAZ-BA	SCSI interface module

Connecting and Using an Alternate Terminal

The system has the ability to display console and diagnostic messages, either on the graphics monitor, or on a VTxxx-type terminal connected to the RS232 port located in the rear of the unit. At power-up, the system looks for a keyboard plugged into the SERIAL.bus. If one is found, the system will route all the console messages to the graphics monitor. If no keyboard is detected on the SERIAL.bus, all text will be sent directly to the RS232 port.

Connect the terminal to the 25 pin RS232 connector, and set the terminal as described in Table 5-3. Notice that these settings are the standard default settings for Digital terminals. Some terminals may use the modular MMJ telephone style connectors. If this is the case, an adapter (Part #H8571-A) can be used to make the connection.

Table 5-3. Alternate Terminal Settings

Setting	Value
Baud rate	9600
Bits	8
Parity	None
Stop bits	1
Flow Control	Xon/Xoff
Mode	Full duplex

Once the terminal is connected to the RS232 port, disconnect the keyboard from the serial port before turning on the system power. This will insure that the text is routed to the terminal and not to the graphics connector.

Use the error messages displayed on the terminal to resolve failures and re-establish video on the graphics monitor.

After establishing video on the graphics monitor, a terminal can be used as a console device. The default power-up console assignment can be defeated by issuing the following console command:

```
>> setenv console s
```

Switch the console back to a graphics monitor with the command:

```
>> setenv console *
```

Some users may configure their systems with several graphics monitors to take advantage of the systems multi-screen capabilities. After the system detects the SERIAL.bus keyboard, it selects the monitor connected to the video controller with the lowest TURBOchannel device number as the default console. Since the base video operates on slot 3, the system will select the controller located in slot 0 or 1 as the default console device.

Minimum Configuration Troubleshooting With the System LED

The most effective way to use the system LED to troubleshoot is to strip the unit to its bare essentials. Such a minimal system consists of the CPU module, base module, and the power supply.

To reduce the unit to a minimum configuration, disconnect all cables in the rear of the system. Then attach the Ethernet loopback connector (Part #12-22196-02) and the SCSI terminator (Part #12-29635-01). Then remove all TURBOchannel options that may be installed as well as all memory modules.

You can then run the power-up self-test and interpret the flashing system LED to isolate and correct problems with this core system.

The keyboard and video sub-system can then be reinstalled. If necessary the power-up self-test can again be used to isolate and correct problems with the video.

Once the video is re-established, the remaining components can be reinstalled one at a time. If a problem occurs at this point, the error messages displayed on the screen can be used to resolve the problem.

More experienced users may not need to strip the system down to a minimum configuration. However, troubleshooting a fully configured workstation with the system LED generally requires a significant amount of knowledge of system operation.

Understanding the Power-up Self-Test Sequence and the LED Display

A single LED, mounted on the front of the power supply, flashes several times during system power-up. These flashes indicate how far the unit has progressed through the power-up sequence.

By interpreting these flashes you can determine which pieces of the unit are operational and then deduce which modules are not functional. It is important to watch the LED closely during the entire power-up sequence.

Note: *The front-mounted system LED is a new feature which has been introduced with the Personal DECstation that is designed as a diagnostic substitute for the traditional eight-segment LED display located on the rear of DECstation models 100, 200, and 240. It is **not** a DCOK LED.*

The Power-up Sequence

- Immediately after the system power comes on, the firmware will begin to execute code.

2 flashes will be displayed. This indicates that the power supply works, and that portions of the CPU and Base Module are operational.

- The LED then goes dark for about 30 seconds. During this time period the console code will be loaded into the memory. The system will then identify what hardware is present, and select which device is to be the default video console.

Continuous flashes will be displayed if a problem is encountered using the memory contained on the base module.

When the system reaches the point where text will be displayed on the default console device, the LED will flash one of two patterns:

2 flashes indicate that a graphics monitor has been selected.

3 flashes indicate that the RS232 port has been selected.

- The LED then goes dark a second time for about 30 seconds. The power-up self-test diagnostic is executed. At this point, status and error messages will be displayed on the console device.

When the self-test is complete, the LED will be **turned on and will stay on**.

Figure 5-1, Table 5-4, Table 5-6, and Table 5-5 contain additional information which may be useful when troubleshooting with the system LED.

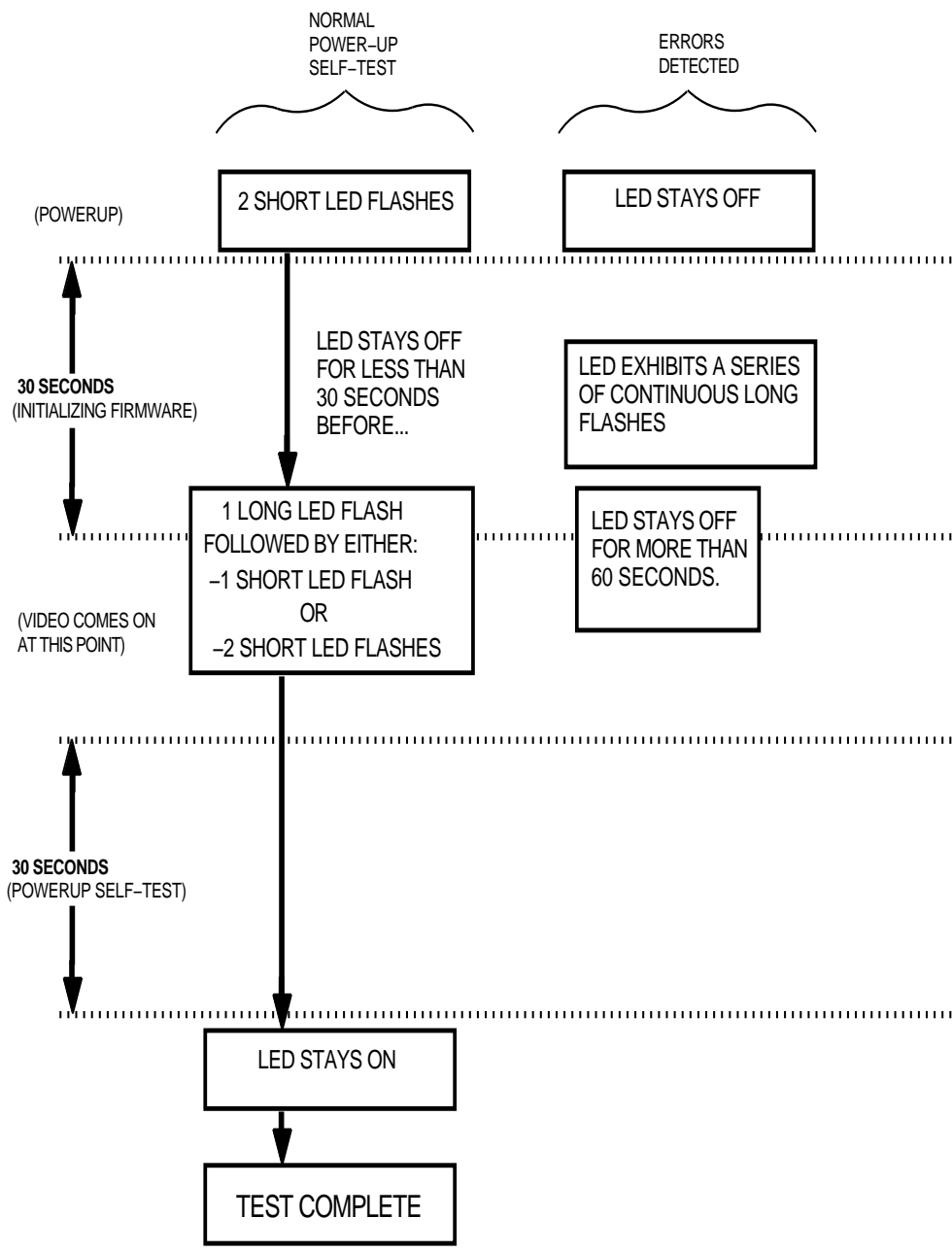


Figure 5-1. Flash Sequence of the Diagnostic LED Indicator

Table 5-4. Interpreting Diagnostic LED Indicator at Power-Up

LED Function	Meaning	Corrective Action
2 short flashes at power-up	Power-on indication set immediately after system firmware begins execution. Indicates that: 1. System is receiving DC power at correct voltage levels. 2. Instructions can be fetched from system ROM via TURBOchannel Read operations. 3. System processor is able to execute non-cached instructions. 4. Processor can perform TURBOchannel write operations to a register in the I/O CTRL ASIC that drives the power supply LED.	None.
LED stays off at Power-up	Power supply, CPU, TURBOchannel option, system ROM, or I/O CNTL ASIC could be at fault.	Check power supply, swap CPU, swap base module.

Table 5-5. Interpreting Diagnostic LED Indicator During 2nd 30 Seconds Following First Video (Power-Up Self-Test Period)

LED Function	Meaning	Corrective Action
LED stays on continuously	Firmware test sequence has completed without detection of any errors.	None.

Table 5-6. Interpreting Diagnostic LED Indicator During First 30 Seconds After Power-Up (Firmware Initialization Period)

LED Function	Meaning	Corrective Action
Off for less than 30 seconds after first two flashes.	Power-up self test in progress.	None
Off for more than 30 seconds after first two flashes.	Some failure has been detected during the power-up self-test that could not be isolated.	Swap base module and/or CPU daughter card.
A continuous series of long flashes occurs.	Memory failed during firmware initialization.	Remove all memory SIMMs and retest to determine if the memory on the base module or on a SIMMs is faulty.
One long flash followed by 1 short flash	<i>kbd</i> (SERIAL.bus keyboard input) and <i>crt</i> have been selected as the firmware console devices: the system is configured as a workstation.	None.
One long flash followed by two short flashes	<i>tty</i> (Serial line input/output) connected to the Personal DECstation's RS232 port has been selected as the firmware's console device: system is configured as a server.	None, unless you were expecting a workstation configuration, in which case you should troubleshoot the SERIAL.bus keyboard.

Note: *Actual test time will vary depending on the memory and the TUROBchannel options which have been installed.*

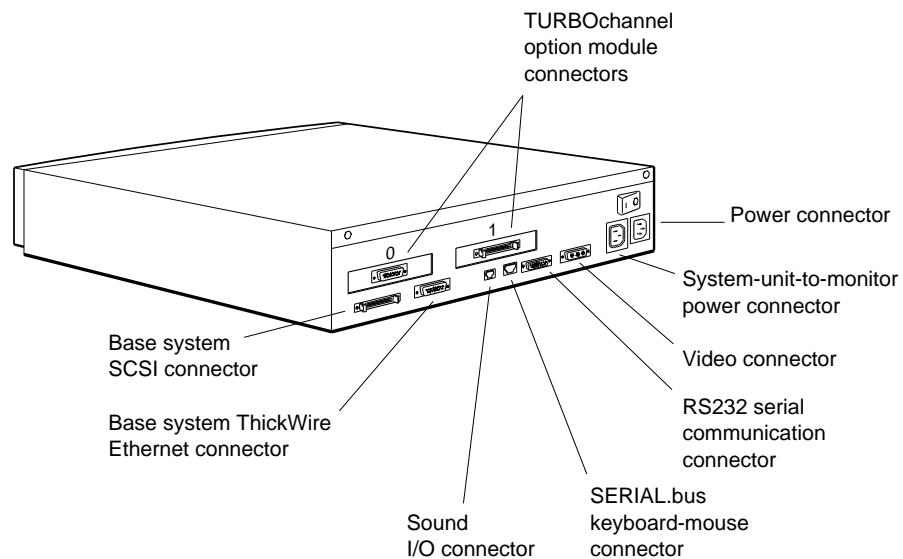
What to do when the Power-up LED Indicator is Inactive

Sometimes the single power-up indicator of the Personal DECstation/DECsystem is itself not working.

When the powerup indicator is inactive, it means one of the three following parts are defective:

- power supply
- CPU daughter card
- base system module

To determine if the power supply is defective, turn on the system power and place an Ethernet loopback connector (12-22196-01) on the Ethernet port (shown as *Base System Thickwire Ethernet connector* in Figure 5-2. The LED in the loopback connector will glow when the power supply is generating correctly regulated DC power.



WSMXM005

Figure 5-2. External System Unit Connectors

If the loopback connector LED does *not* glow, replace the power supply.

If the loopback connector *does* glow and yet the power-up indicator LED is still inactive, the problem is with either the CPU daughter card or the system module. Retest the system after replacing the CPU daughter card (easiest to remove) followed by the system module, if necessary.

How to Troubleshoot the R4000 CPU Module LEDs

Two LED's on the Daughter card can provide a quick indication of the working condition of the daughter card and system module. If either of the LED's are off then replace the daughter card.

ROM-Based Console and Diagnostics

This chapter discusses the following topics:

- Troubleshooting techniques
- On-line help
- Console commands available for troubleshooting
- Configuration display as a troubleshooting aid
- Effects of Environment variables on system operation
- Interpreting status and error messages displayed by the system
- Diagnostic scripts and tests for troubleshooting specific problems

Console Commands

The console commands are used to obtain more detailed information about how the hardware is configured and about how various system parameters, called environment variables, are set. The use of diagnostic scripts and tests direct the hardware to exercise a specific subsystem.

Once the power-up self-test has completed, the console prompt **>>** is displayed on the screen. Table 6-1 contains a list of console commands available. This table has been organized by function. For a detailed description of each command, look in Appendix D.

The **R>** prompt is a secure prompt that signifies the **passwd** command has been used to password-protect the console. Once the console has been secured, only the **boot** and **passwd** commands will be processed.

This security feature was added for systems that are deployed on secure government sites. Using the **passwd** feature is not recommended unless there is a compelling reason to do so.

The console command syntax has the ULTRIX look and feel. Some commands like **ls** and **cat** are very similar to those used in ULTRIX. Even the bootpath strings look similar to the path strings used by ULTRIX for file specifications.

Table 6-1. Console Commands

Command	Description	Type
?	Displays a list of console commands and formats.	Displays system information.
cnfg	Shows HW configuration.	Displays system information.
printenv	Shows environment variables.	Displays system information.
erl	Shows error history.	Displays system information.
setenv	Sets environment variables.	Sets system parameters.
unsetenv	Removes environment variables.	Sets system parameters.
passwd	Sets console password.	Sets system parameters.
t #/?	Lists test for slot #.	Lists diagnostic routines.
ls #	Lists scripts.	Lists diagnostic routines.
cat Script	Lists test contained in a script.	Lists diagnostic routines.
t #/t_name	Executes a test on slot #.	Runs diagnostics.
script_name	Executes a script.	Runs diagnostics.
sh	Executes a script with options.	Runs diagnostics.
boot	Boots the system.	Operational commands
init	Initializes the system.	Operational commands
d	Deposits data into memory.	Miscellaneous commands
e	Examines memory locations.	Miscellaneous commands
go	Go to an address.	Miscellaneous commands
script	Creates a temporary script of console commands.	Miscellaneous commands

Similarity to Other DS5000 Systems' Console Commands

The console implementation of all DS5000 machines is defined by the TURBOchannel firmware specification. As a result, the basic command syntax is the same across the entire family of systems.

There are few differences between the Personal DECstation/DECsystem and other members of the DS5000 product set. The Personal DECstation/DECsystem introduces some new hardware features, which are described in Table 6-2. Some new diagnostic scripts and tests have been added to support these new hardware features.

Some of the names of familiar scripts and tests may have changed. This is a result of a continual engineering effort to enhance the diagnostics with the release of each new machine.

Appendix D contains a detailed treatment of all the console commands.

Throughout this chapter there are output listings for various commands. In all cases, this output was generated for a single hardware configuration. This configuration is a Personal DECstation/DECsystem with a Mid-3D graphics option. These output listings have been selected to show you how to obtain various types of information that are useful when troubleshooting a problem system.

On-line help

The **?** command lists the available console commands. This display shows the parameters, arguments, and qualifiers commands can take on.

Because the console is ROM-based, there are practical limits to the code space available for the help facility. As a result, the on-line help does not provide a detailed description of each command. However this display is a good memory aid, for those who are familiar with the DS5000 console.

```
>> ?
?[cmd]
boot [[-z #] [-n] #/path [ARG...]]
cat SCRPT
cnfg [#]
d [-bhw] [-S #] RNG val
e [-bhwcdoux] [-S #] RNG
erl [-c]
go [ADR]
init [#] [-m] [ARG...]
ls [#]
passwd [-c] [-s]
printenv [EVN]
restart
script SCRPT
setenv EVN STR
sh [-belvS] [SCRPT] [ARG..]
t [-l] #/STR [ARG..]
unsetenv EVN
```

Configuration Display

One of the most useful troubleshooting tools is the **cnfg** command. This command displays hardware configuration information. Early in the power-up sequence a data structure is created in the memory with an entry for all hardware contained in the system. The **cnfg** command reads this data, formats it, and generates a report that is displayed on the video console.

By comparing the hardware physically attached to the system with the report generated by the **cnfg** command, you can often identify portions of the hardware that are not operational. There are many reasons for hardware not being configured into the system. They range from improper installation, to defective hardware. Further investigation will determine the exact cause of the problem.

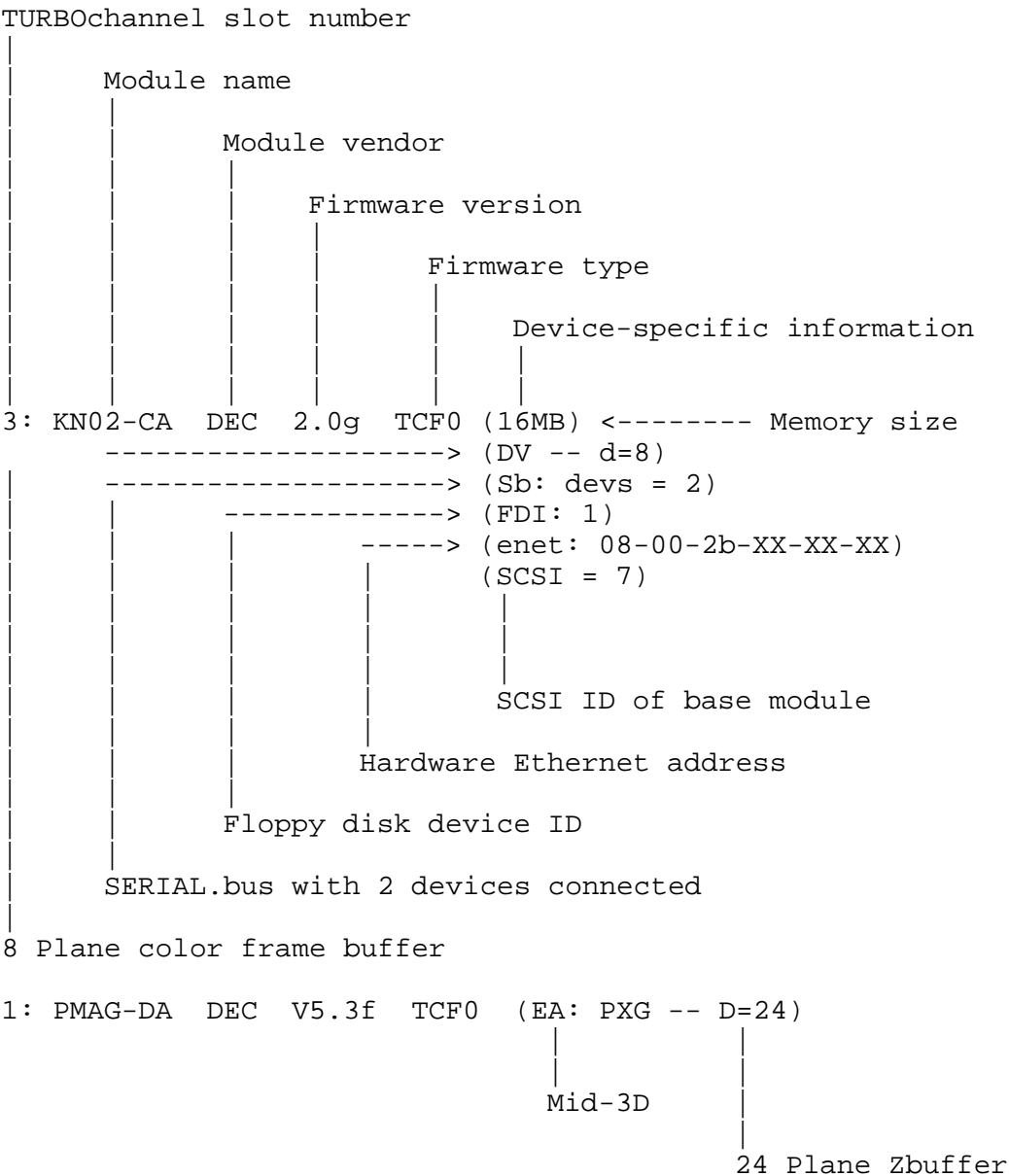
The command shown below is used to display configuration information for the entire system.

```
>> cnfg
```

Example: cnfg Command output The example below shows a typical output screen from invoking the *cnfg* command on a system with the R4000-based CPU module installed.

```
>>cnfg
3: KN04      DEC      V1.0a      TCFO  ( 32 MB)
                                     (enet: 08-00-2b-2d-84-c7)
                                     (SCSI = 7)
2: PMAD-AA   DEC      V5.1f      TCFO  (enet: 08-00-2b-0f-41-32)
1: PMAZ-AA   DEC      V5.1c      TCFO  (scsi: = 7)
0: PMAG-BA   DEC      V5.2b      TCFO  (CX -- d=8)
```

The sample **>>cnfg** command output displayed below shows a typical R3000-based system output, and has been annotated to identify the key pieces of information.



From the example **>>cnfg** display it can be determined that

- Two devices are connected to the SERIAL.bus.
- The SCSI ID of the base module controller is 7.
- 16MB of memory are installed.
- A floppy disk drive is connected to the base module.
- A mid-3D graphics accelerator is installed in slot 1.
- If the R4000 CPU is installed, the module name is KN04-xx
vice KN02-xx

The cnfg x Command The **cnfg** command can also display more detailed configuration information for a specific slot using the following format:

```
>> cnfg #
|
-----
# is the slot number (0,1,3)
```

Example: cnfg 3 Output Screen The >> **cnfg 3** command output displayed below has been annotated to identify the key pieces of information. This output represents an R3000 CPU-based system.

```
3: KN02-CA  DEC   2.0g   TCF0   ( 16MB)
                                   (DV -- d=8)
                                   (Sb: devs = 2)

Name          Rev.          Vendor
=====
LK501-AA      AV1.3          DEC
VSXXX-BB      AV1.2          DEC

                                   (enet: 08-00-2b-XX-XX-XX)
                                   (FDI: 1)
                                   (SCSI = 7)

-----
DEV   PID          VID   REV   SCSI DEV
=====
rz1   RZ25  (c) DEC  DEC   0500  DIR
rz2   RZ55  (c) DEC  DEC   0900  DIR
rz3   RRD42 (c) DEC  DEC   1.4a  CD-ROM

dcache ( 64 KB), icache ( 64 KB)
mem( 0): a0000000:a03ffffff ( 4 MB)
mem( 1): a0400000:a07ffffff ( 4 MB)
mem( B): a0800000:a0ffffff ( 8 MB)
```

From the >> **cnfg 3** display, more detailed information can be obtained about the hardware contained in and attached to the base module. It can be determined that:

- An LK501 keyboard and a VSXXX-BB mouse are connected to the SERIAL.bus.
- Three drives are connected to the base module SCSI port. These drives are an RZ25 winchester disk, an RZ55 winchester disk, and an RRD42 CDROM. The SCSI device IDs for these drives are 1, 2, and 3 respectively.
- Four MS01-AA modules are installed on the base module.

The screen below represents the output display result of the **cnfg 3** command with the R4000 CPU module installed.

```
>>cnfg 3
3: KN04      DEC      V1.0a      TCF0  ( 32 MB)
                                     (enet: 08-00-2b-2d-84-c7)
                                     (SCSI = 7)

-----
DEV  PID                                VID      REV      SCSI DEV
=====
rz1  RZ25  (c) DEC  DEC  0500  DIR
rz2  RZ55  (c) DEC  DEC  0900  DIR
rz3  RRD42 (c) DEC  DEC  1.4a  CD-ROM

cache: I( 8 KB), D( 8 KB), S(1024 KB);  Scache line (32 bytes)
processor revision (3.0)
mem( 0): a0000000:a07ffffff ( 8 MB)
mem( 1): a0800000:a0ffffff ( 8 MB)
mem( 2): a1000000:a17ffffff ( 8 MB)
mem( 3): a1800000:a1ffffff ( 8 MB)
```

Note in the above example that the screen output is the same as that of an R3000 based system, with two exceptions:

1. *cache* line of the R4000 screen changes to Instruction, Data, Secondary and Secondary cache line size on an R4000-type system.
2. An additional processor revision information line just below the *cache* line.

TURBOchannel Slot Number Assignments

Table 6-2 contains a list of the TURBOchannel slot number assignments for Personal DECstation's hardware subsystems. This information is a useful reference while interpreting various console displays.

Table 6-2. Slot Numbers in Commands and Messages

Slot	Hardware Identified
0	Option module in slot 0 (farthest from the power supply)
1	Option module in slot 1
2	Reserved
3	Base system hardware, which includes <ul style="list-style-type: none">– System module– CPU module– Base system RAM– SIMMs– Base system video controller– Base system SCSI controller– Base system Ethernet controller– SERIAL.bus (keyboard and mouse)– Base system serial communications controller

Environment Variables

There are times when a system does not function as expected, yet all the hardware is operational.

One source of such problems can be improperly set system parameters called environment variables. For example, we may expect a system to automatically reboot itself after a power failure. If it does not, the cause could be an incorrect setting for either the **boot** or the **haltaction** environment variables.

```
>> printenv
boot=3/rz2/vmunix -a
testaction=t
haltaction=b
more=24
enet=08-00-2b-xx-xx-xx
console=
osconsole=1,3
#=3
```

From the information in this environment variables listing, it can be determined that:

- The system will boot into multiuser mode from an RZ drive with the SCSI device ID 2.
- When this system is halted, or if a power failure occurs, the machine will automatically reboot itself.
- On power-up, this system will run a thorough version of the power-up self test diagnostic.
- Console messages will display on the monitor attached to the graphics module in TURBOchannel slot 1.

Environment variables are stored in a data structure in memory. The **printenv** command reads this data and formats it to generate the report that is displayed on the video console. The operating system uses call back routines to read the environment variables and obtain needed hardware information. In addition, the diagnostic scripts read and write the environment variables during their normal operation.

Modifying environment variables with the **setenv** command can also be a useful troubleshooting tool. In Chapter 5, we used the console environment variable to redirect the console error messages to an alternate terminal. Setting the testaction variable to thorough, will cause extended testing to be conducted during power-up self test.

Changing the haltaction variable from boot, where it is most often set, to halt will keep the system from automatically booting after power-up. Having the system continually boot itself during testing can be annoying.

Table 6-3 contains a list of environment variables with some additional descriptive material.

Table 6-3. Environment Variables

Variable	Argument Summary	Type
boot	boot path	User-defined
haltaction	b=boot,h=halt,r=reset	User-defined
testaction	q=quick,t=thorough	User-defined
console	s=RS232, other=CRT	User-defined
more	number lines /video screen	User-defined
enet	08-00-2b-xx-xx-xx	System-defined
#	Slot 0,1,3	System-defined
osconsole	video slot,keyboard slot	System-defined

Several of the environment variables are system-defined. Although the user cannot modify these settings, these variables do contain useful information. For example, **osconsole=1,3** in the environment variable display on, indicates that console text is displayed through the video controller in TURBOchannel slot 1 and that console input is obtained from a keyboard attached to TURBOchannel slot 3 on the base system module.

Interpreting Error Messages

When a test detects an error, a formatted message is displayed on the console. The example below shows an error message that indicates a problem has been detected with the SCSI control portion of the base module.

Error Message Format:

?TFL slot_number/test_name (n:description) [module]

Example: Console Error Message

There were three key pieces of information that led to this conclusion. The failure was detected on TURBOchannel slot 3. The problem was with the SCSI subsystem and was, specifically, the controller. Finally, the diagnostic identified the module as the KN02-CA base module.

Example:

```

Name of the test
-----
|
|   Name of subtest that failed
|   -----
|   |
|   |   Descriptive failure info
|   |   -----
|   |   |
?TFL 3/scsi/cntl (3: cntxfr) [KN02-CA]
|   |   |
|   |   |   ID of failed module
|   |   |   -----
|   |   |
|   |   |   -----
|   |   |   Number of subtest that failed
|   |   |   -----
-----
TURBOchannel
slot number

3= Base module, CPU, and memory
0= TURBOchannel option in slot 0
1= TURBOchannel option in slot 1

```

Tables 6-4, 6-5, and 6-6 contain reference material that is useful when interpreting error messages.

Appendix E contains detailed descriptions of all diagnostic errors. Below is an excerpt from this appendix for the SCSI error at hand. This specific failure was a read/write mismatch that occurred when data was passed to the SCSI controller.

SCSI controller test error messages have the form

? TFL: 3/scsi/cntl (*code*: *description*)

- ?TFL 3/scsi/cntl indicates that the SCSI controller test failed.
- *code* represents a number that indicates which portion of the test failed.
- *description* represents additional information that describes the failure.

Table 6-4 lists error descriptions used in SCSI controller test error messages.

Table 6-4. SCSI Controller Error Codes and Descriptions

Error Code	Description
1: rd cnfg	Values written to and read from configuration register did not match.
2: fifo flg	First in, first out (FIFO) load and FIFO flags did not match.
3: cnt xfr	Write and read operation on TCL register reported a mismatch.
4: illg cmd	Command was illegal and did not generate an interrupt.
5: int reg	Controller cannot clear internal interrupt register.
6: rd cnfg	Mismatch occurred when reading the write/read configuration register.

Table 6-5. Diagnostic Tests for Base System

Test	Subsystem	Problem Module
ab	SERIAL.bus	Base module, keyboard, or mouse
cache	CPU	CPU daughter card
crt	Video frame buffer	Base module
epm	EEPROM	Base module
fdi	Floppy disk	Base module or floppy drive
fpu	CPU	CPU daughter card
isdn	ISDN and sound	Base module, ISDN peripherals
mem	Memory	Base module, memory SIMMs
ni	Ethernet	Base module, Ethernet card, Ethernet peripherals
rtc	Real time clock	Base module
scc	RS232	Base module, RS232 peripherals
scsi	SCSI	Base module, SCSI card, SCSI peripherals
scache	Secondary Cache	CPU daughter card

Table 6-6. Identification Codes for Selected TURBOchannel Options

Module ID	Description
PMAG-AA	MX mono frame buffer
PMAG-BA	CX color frame buffer
PMAG-CA	PX 2D graphics accelerator
PMAG-DA	PXG 3D graphics accelerator
PMAG-FA	PXG turbo graphics accelerator
PMAGB-BA	HX smart frame buffer
PMAG-JA	TX true color frame buffer
PMAD-AA	NI ThickWire Ethernet
PMAZ-BA	SCSI interface module

Table 6-7 contains a summary of all the error messages the console will display.

Console command error messages

Table 6-7. Console Command Error Messages

Error Message	Meaning
?EV:ev_name	The specified environment variable does not exist.
?EVV:value	The specified environment variable value is invalid.
?IO:slot_number/device	An I/O device reported an error. <i>slot_number</i> represents the I/O device slot number, and <i>device</i> represents an additional message about the error.
?IO:slot_number/device	The module in the slot represented by <i>slot_number</i> does not recognize the device represented by <i>device</i> .
?PDE3:slot_number	The module in the slot represented by <i>slot_number</i> contains an early version of firmware. The ROM chip must be upgraded.
?SNF:script	The system did not find the script that was to be run.
?TXT:	The name specified in the script command is not a valid script name.
?STX:usage	A console command contained a syntax error. The usage parameter lists the correct syntax.
?STX:error	A console command contained a syntax error. The error parameter lists the incorrect portion of the command.
?TFL:slot_number/ test	A test failed. <i>slot_number</i> represents the slot number of the module that reported the error, and <i>test</i> represents the name of the failed test.
?TNF:	Test not found. The test name was probably typed incorrectly.

Console exception messages

It is important to remember that the console commands are programs that execute on the hardware we are attempting to diagnose.

Occasionally, a hardware problem will cause the console code to fail. When this occurs the console is said to have taken an exception. In such a case, the following type of message is displayed. Notice that this display has a different format than test error messages.

By interpreting the register values listed in the console exception message, you can determine what caused the exception. The R3000 console exception message has the following format:

```
? PC:  address
? CR:  cause
? SR:  status
? VA:  virtual address
? ER:  error address
? MER: memory error
```

- *address* represents the address of the exception instruction.
- *cause* represents the value in the cause register.
- *status* represents the contents of the status register.
- *virtual address* represents the virtual address of the exception.
- *error address* represents the contents of the error address register.
- *memory error* represents the value in the memory error register.

The following is a typical R3000 console exception message:

```
? PC:  0x451<vtr=nrml>
? CR:  0x810<ce=0,ip4,exc=AdEL>
? SR:  0x30030000<cu1,cu0,cm,ipl=8>
? VA:  0x451
? ER:  0x100003f0
? MER: 0x2000
```

The following is a typical R4000 console exception message:
In each entry, the information in brackets (<>) is the decoded version of the hexadecimal value that precedes it.

```
???  
? PC:  0xbfc0cd60 <vtr=TLBM>  
? CR:  0x00000008 <CE=0,EXC=TLBL>  
? SR:  0x30010002 <CU1,CU0,DE,IPL=8,MODE=KNL,EXL>  
? CFG: 0x10410243 <SB=8W,SC=Y,IC=8K,DC=8K,IB=4W,DB=4W,K0=CNC>  
? VA:  0x00000000  
?  
? MB_CS: 0x00008000 <MSK=0,EE,ECC=0>  
? MB_INT: 0x001f0000 <>
```

Diagnostic Tests

There are hundreds of diagnostic scripts and tests available for troubleshooting. With these diagnostics, the entire machine or specific subsystems can be exercised. In this section we show you how to identify the diagnostic routines that will isolate the source of the particular problem.

The approach presented in this guide is to start with a fairly general diagnostic that covers the entire system. After this, diagnostics that do a progressively more specific job of isolating the problem can be used.

The diagnostics can be divided into two categories: scripts and tests. Tests are the most specific diagnostics. Hundreds of individual tests are available. Each has been designed to exercise a specific portion of the hardware.

Diagnostic scripts are a collection of tests selected to exercise a segment of hardware. These scripts are much like ULTRIX script files.

When the power-up self-test is run, a script named either pst-q or pst-t is executed. This is a general script that exercises the entire machine. The tests in the script have been selected to ensure that all of the key components are tested while keeping the total test time to a minimum.

From the error messages contained in the pst-q script report, it can be determined which subsystem of a specific TURBOchannel option is having difficulty. The ls command can then be used to identify what scripts are available to exercise the affected hardware.

The cat command can be used to ascertain what a particular script does. The cat command displays a list of the tests contained in the script. If a more specific exerciser is needed, the t command will list all of the individual tests for a specific TURBOchannel slot. Finally, Appendix E can be consulted for a description of what each test does.

Selecting diagnostic routines

If the pst-q power-up self-test script identifies an error, a more specific test may be needed to isolate the problem. In the following example, a problem was detected with the PMAG-EA Mid-3D graphs accelerator in slot 1. The ls command is used to list the scripts that will exercise this module.

```
>> ls 1
      28 1 cnfg -> code
      28 1 init -> code
      28 1 t -> code
     132 1 cnsltest
     216 1 pst-q
     234 1 pst-m
     450 1 tstall
     232 1 pst-t
    7664 1 crt*
   54544 1 code*
```

The cat command is then used to find out what tests the cnsltest script contains.

```
>> cat 1/cnsltest
t ${#}/sram
t ${#}/stic
t ${#}/intr-dma
t ${#}/utl/mkcnfgtbl
t ${#}/stamp
t ${#}/vdac
t ${#}/vdacsig
```

Note: This example was chosen only because the output produced by the ls, t, and cat commands was small enough to be presented easily.

The `cnsltest` script runs seven tests, each of which exercises a specific portion of the 3d graphics module.

There are additional tests available. The `t` command can be used to list all the tests available for slot 1.

```
>> t 1/?
1/
sram
sramfill      lo[0]  <1fffc/0>  hi[10000]  <1fffc/0>
stic
intr-dma
vdc
vdacsig
stamp
shade
fb            buff[*] <0/3>
utl/pattnrs   snlstp[0] <0/1>
utl/mkcnfgtbl
utl/i890exc   md[f] <k/s> cd[*] <j/d> loc[r] <sys/rom> rn[e] <e/n>
i860/stic
i860/sticintr
i860/intrhost
i860/dma      type[r] <w/r> cnt[80] <1/80>
i860/fbfill   patt[ffffff] <0/ffffff> buff[*] <0/3>
```

There are a total of seventeen tests for the mid-3d graphics module. The `cnsltest` script uses seven of them.

At times, a single test may be more useful than a script. One such test is `utl/pattnrs 1`. It displays the alignment patterns that are used to adjust a monitor's display. By using the `1` argument, striking any key will move the display from one pattern to the next.

The detailed descriptions of all the base module, CPU, and memory tests are described in detail in Appendix E. A description of the TURBOchannel option tests can be found in the *TURBOchannel Maintenance Guide*.

ULTRIX Error Reporter and System Exerciser

This chapter discusses the following topics:

- Troubleshooting techniques
- On-line help
- ULTRIX error logs
- Generating error reports
- Interpreting part one of the error report
- Interpreting part two of the error report
- Running the system exercisers
- Exercising the file system and memory
- Exercising peripherals
- Exercising the network and communication systems

Troubleshooting Techniques

In this chapter we see how the ULTRIX error report formatter (UERF) and the ULTRIX system exerciser utilities are used to resolve problems of more subtle origins. These problems may be intermittent and can involve hardware-software interaction.

The error reporter and system exercisers are best used in pairs. First, analyze the reports UERF produces and develop a hypothesis as to which subsystem has a problem. Then use the system exerciser to generate activity on the suspect hardware

subsystem. Any errors the exerciser induces will be recorded by the ULTRIX error logging facility. UERF can again be used to retrieve the failure information from the error log. It may be necessary to repeat this procedure several times to narrow in on the origin of the problem.

The material presented in this chapter assumes the reader has a working knowledge of the ULTRIX operating system.

ULTRIX Error Logs

The ULTRIX operating system contains an error logging facility that continuously monitors the status of the hardware. This facility records significant events in a log file. Some events such as, disk and Ethernet traffic, indicate operational status. Other events, such as memory exceptions, indicate hardware failures.

The log file's format has been optimized to enable the error logger to record events and not degrade system performance. As a result, it is difficult for people to decipher the file. The ULTRIX error report formatter (UERF) utility reads the log file and produces easily read reports.

The error logger records large amounts of data. Generally, we are only interested in selected pieces of information. With UERF, we can sort through the log file to filter out the specific events in which we are interested.

Generating error reports

Error reports are produced by running the **uerf** program, which is generally located in the `/etc` directory.

The report shown here contains the last memory error the system encountered. It was generated by using the `-R` and `-r` options. The following ULTRIX command was used to create this report:

```
/etc/uerf -R -r101 | more
```

```

----->
-----EVENT INFORMATION-----

P   EVENT CLASS                      ERROR EVENT
A   OS EVENT TYPE                    101. MEMORY ERROR
R   SEQUENCE NUMBER                  5697
T   OPERATING SYSTEM                 ULTRIX 32
1   OCCURED/LOGGED ON                Tue Oct 23 12:18:21 1990 EDT
   OCCURED ON SYSTEM                 nettot.nac.d
   SYSTEM ID                          x82012001 HW REV: x1
                                       FW REV: x20
                                       CPU TYPE: R3000A
   PROCESSOR TYPE                     DECSTATION XXXXX
----->

----->
-----UNIT INFORMATION-----

UNIT CLASS                      MEMORY
UNIT TYPE                      PMAX MEMORY
ERROR SYNDROME                 MEMORY PARITY ERROR

-----MEMORY INFO-----

P   MEMREG                          x56410000 HARD CNT 0.
A                                       SOFT CNT 0.
R                                       TRAN CNT 65.
T                                       DATA BIT IN ERROR
2                                       HIGH BYTE IN ERROR
                                       MEMORY PARITY ERROR
                                       SIMM # 5

   PHYSICAL ADDR    x00849E0C
   EPC              x0040CE0C
   VIRTUAL ADDRESS  x0040CE0C

-----ADDITIONAL INFO-----

CONTROLLER NO.      1.
ERRS ON THIS ADDR   1.
----->

```

On-line help

The ULTRIX man pages for UERF contain a brief overview of the utility along with definitions of the various processing and selection options available to generate reports. The following ULTRIX command is used to display the man pages:

man uerf

In addition, the UERF program has a help option, -h. The following ULTRIX command will display the help information shown below:

/etc/uerf -h | more

UERF HELP

uerf - Processes all events in the default input file

PROCESSING OPTIONS

-h	- Displays the help screen.
-f input_filename	- Processes events from the specified input file.
-n	- Processes events from kernal errorlog buffer immediately.
-H ascii_host_name	- Processes events for the specified hosts.
-R	- Processes events from the errorlog file in reverse order.
-S	- Displays a summary of the selected events.
-t time_range	- Processes events in a specified time range.
-o output_format	- Formats the output in brief, full, or terse.
-x	- Excludes selected events (shown below) from output.
-Z	- Displays the entry in hex format.

SELECTION OPTIONS - Events may be selected as follows:

-A adapter_list	-D disk_devices	-M mainframe_type
-O os_specific	-T tape_devices	-c class_of_events
-e error_types	-r record type	-s sequence number

***** For a detailed list of options see uerf (8) *****

ULTRIX error log event types

The information contained in Table 7-1 is a useful reference when deciphering error reports.

These codes are also the parameters used with the UERF -r option to generate reports for specific types of events.

Table 7-1. Event Types

Code	Event Type
100	Machine check
101	Memory error
102	Disk error
103	Tape error
104	Device controller error
105	Adapter error
106	Bus error
107	Stray interrupt
108	Asynchronous write error
109	Exception or fault
113	CPU error and status information
130	Error and status register
200	Panic (bug check)
250	Informational ASCII message
251	Operational message
300	System startup message
310	Time change message
350	Diagnostic information

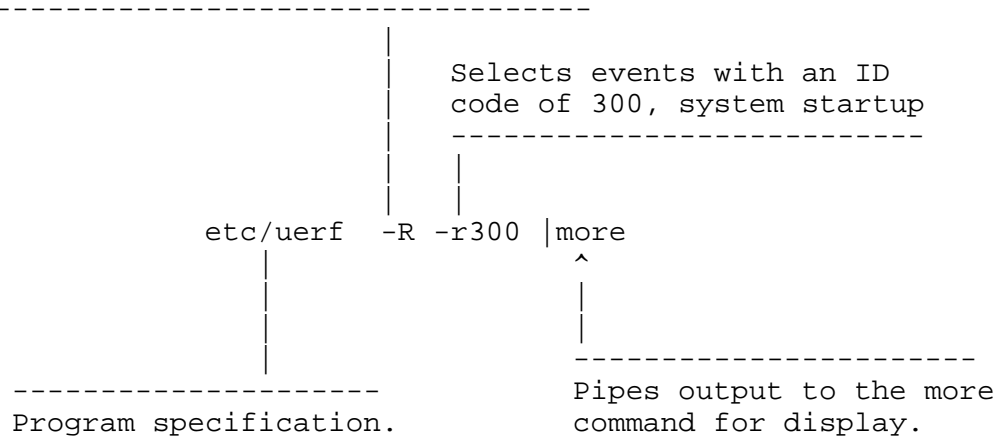
Interpreting part one of the error report

The first part of each ULTRIX error log contains the event information. This includes the error type and the system conditions in effect when the error occurred. The first part of all error reports has the same basic format regardless of event type.

Sample error report

The error report on the next page was produced with the following command:

Displays events in reverse order,
last event first, first event last.



The first part of all ULTRIX error logs is similar to this example:

```

P A R T 1
----->
      ----- EVENT INFORMATION -----
      EVENT CLASS                      OPERATIONAL EVENT
      OS EVENT TYPE                    300.  SYSTEM STARTUP
      SEQUENCE NUMBER                  0.
      OPERATING SYSTEM                 ULTRIX 32
      OCCURRED/LOGGED ON               Tue Oct 24 09:06:16 1989
      OCCURRED ON SYSTEM               GRANITE
      SYSTEM ID                        x82020120  HW REV: x20
                                           FW REV: x1
                                           CPU TYPE: R2000A/R3000
      PROCESSOR TYPE                   KN02/R3000
----->

```

- EVENT CLASS indicates the category of the error. The two event class categories are operational events and error events.
 - Operational events are changes in system operation that are not errors.

- Error events are actual errors in system operation.
- OS EVENT TYPE describes the type of error or event recorded in the log. Table 7-1 lists the operating system event types and their codes.
- SEQUENCE NUMBER indicates the order in which the system logged the event.
- OPERATING SYSTEM indicates the system's version of ULTRIX.
- OCCURRED/LOGGED ON indicates when the error occurred.
- OCCURRED ON SYSTEM indicates the individual system that reported the error.
- SYSTEM ID includes several listings:
 - The first number is the system ID.
 - HW REV indicates the system hardware revision number.
 - FW REV indicates the system firmware revision number.
 - CPU TYPE indicates the type of CPU used in the system.
- PROCESSOR TYPE indicates the type of processor chip the system uses.

Interpreting part two of the error report

The second part of each log provides specific information about the error and its source. This is the information we focus on when troubleshooting. The format of part two is specific to the event type.

The next few pages contain error reports for several different error types. An explanation is provided for each.

Memory Error Report

```
***** ENTRY 1 *****
----- EVENT INFORMATION -----
EVENT CLASS                      ERROR EVENT
OS EVENT TYPE                    101. MEMORY ERROR
SEQUENCE NUMBER                  2.
OPERATING SYSTEM                 ULTRIX 32
OCCURRED/LOGGED ON               Tue Jan 22 16:12:54 1991
OCCURRED ON SYSTEM              ft3
SYSTEM ID                       x82030230  HW REV: x30
                                      FW REV: x2
                                      CPU TYPE: R2000A/R3000
PROCESSOR TYPE                   KN02-BA

----->
----- UNIT INFORMATION -----
UNIT CLASS                      MEMORY
UNIT TYPE                      Undefined  x001C
ERROR SYNDROME                  memory parity error
----- KN02BA MEMORY REGISTERS -----
P  MEM REG      xBE000002  HARD CNT 2.
A                                     SOFT CNT 0.
R                                     TRAN CNT 0.
T                                     DATA BIT IN ERROR
2                                     HIGH BYTE IN ERROR
                                     MEMORY PARITY ERROR: HARD
                                     D0-15 ON BANK # x2
PHYSICAL ADDR      x010205F8
EPC                x004255F8
BAD VIRT ADR       x004255F8
----->
```

Analyzing memory error report

The error log example on the previous page records a memory parity error. Consult Appendix F for detailed information about memory registers.

The following list describes the useful information found in the memory error log on the preceding page. The last entry under MEM REG in ENTRY 1 identifies which SIMM has failed.

- The ERROR SYNDROME field indicates that this is a memory parity error.

- The `MEM REG` fields list details about the memory failure:
 - The `CNT` fields list the count of hard errors (errors that recurred on both read and write operations), soft errors (errors that recurred on read, but cleared on write), and transient errors (errors that did not recur on read) errors.
 - The `DATA BIT IN ERROR` statement indicates that a data bit (rather than a parity bit) failed.
 - The `BYTE IN ERROR` can be either high or low.
 - The `MEMORY PARITY ERROR` field lists the type (hard, soft, transient) of error for this entry.
 - `BANK` tells which SIMM failed. The 15 slots are labeled 0 to 14 on both sides of the memory connectors. To replace the bad SIMM, use the bank number to find the correct slot.
- The `PHYSICAL ADDRESS` field identifies the physical address that was being read when the failure was detected. Because the system uses DMA block read operations (more than one word is read in a read operation), the parity error could have occurred in any word in the block. This address points to only the block being read, not necessarily to the bad word.
- `ENTRY 2` occurring prior to a memory parity error indicates that the memory parity error was detected by an I/O device performing a DMA operation from memory.

Error and status register error logs

Error and status register error logs record nonmemory system errors. The following examples show an error and a status event: a CPU write timeout and a bus timeout. In both examples the event type is `PANIC`. Therefore, the initial error log is followed by a related error and status register error log. Each example is followed by an analysis of the information in the specific entry that is relevant to the error.

Error and status register error log: Example 1

The first entry in the following example, labeled ENTRY 2, records a CPU write timeout error. Because the event type is PANIC, it is followed by a related log, labeled ENTRY 3, that records the error and status register values at the time of the error. See Appendix F for detailed information about memory registers.

The useful information in ENTRY 2 and ENTRY 3 is as follows:

- ENTRY 2 records the error.
- The OS EVENT TYPE field indicates the PANIC type.
- The PANIC MESSAGE field indicates that the error is a CPU write timeout.
- ENTRY 3 records the error and status register values at the time of the CPU write timeout error.
- The CAUSE register has no information because the write timeout is an interrupt.
- The BAD VIRT ADR register identifies the address of the timeout.
- The SIR register indicates that the system interrupt was a CPU I/O write timeout.

```

***** ENTRY 2. *****
----- EVENT INFORMATION -----
EVENT CLASS                      ERROR EVENT
OS EVENT TYPE                    200.  PANIC
SEQUENCE NUMBER                  2.
OPERATING SYSTEM                 ULTRIX 32
OCCURRED/LOGGED ON              Tue Feb 26 06:38:28 1991 EST
OCCURRED ON SYSTEM              csselab1
SYSTEM ID                       x82030230  HW REV: x30
                                FW REV: x2
                                CPU TYPE: R2000A/R3000
PROCESSOR TYPE                   KN02-BA
PANIC MESSAGE                    CPU write timeout
***** ENTRY 3. *****
----- EVENT INFORMATION -----
EVENT CLASS                      ERROR EVENT
OS EVENT TYPE                    130.  ERROR & STATUS REGS
SEQUENCE NUMBER                  1.
OPERATING SYSTEM                 ULTRIX 32
OCCURRED/LOGGED ON              Tue Feb 26 06:38:26 1991 EST
OCCURRED ON SYSTEM              csselab1
SYSTEM ID                       x82030230  HW REV: x30
                                FW REV: x2
                                CPU TYPE: R2000A/R3000
PROCESSOR TYPE                   KN02-BA
- ----- KN02BA REGISTERS -----
CAUSE                           x00002800
                                EXCEPTION CODE EXTERNAL INTERRUPT
                                HW INTERRUPT 1 PENDING
                                HW INTERRUPT 3 PENDING
EPC                             x801266FC
STATUS                          xFFFFDC84  CURRENT INTERRUPT STATE DISABLED
                                CURRENT MODE KERNEL
                                PREVIOUS INTERRUPT STATE ENABLED
                                PREVIOUS MODE KERNEL
                                OLD INTERRUPT STATE DISABLED
                                OLD MODE KERNEL
                                HW INTERRUPT 0 ENABLED
                                HW INTERRUPT 1 ENABLED
                                HW INTERRUPT 2 ENABLED
                                HW INTERRUPT 4 ENABLED
                                HW INTERRUPT 5 ENABLED
                                D_CACHE ISOLATED FROM MEMORY
                                CACHE STATE SWITCHED
                                PARITY ZERO SET

```

```

                                CACHE MISS SET
                                CACHE PARITY ERROR SET
                                TLB SHUTDOWN SET
                                BOOTSTRAP EXCEPTION VECTOR SET
BAD VIRT ADR      x10000000
SP                xFFFFDDA8
SSR               x50070F00
                                LANCE DMA DISABLE
                                SCSI DMA ENABLE
                                SCSI DMA WRITE TO MEMORY
                                COMM PORT 2 RECEIVE DMA DISABLE
                                COMM PORT 2 TRANSMIT DMA ENABLE
                                COMM PORT 1 RECEIVE DMA DISABLE
                                COMM PORT 1 TRANSMIT DMA ENABLE
SIR               x00001002
                                CPU I/O - WRITE TIMEOUT
SIRM              x0000F204
                                SCSI INTERRUPT FROM 53C94 ENABLED

```

Error and status register error log: Example 2

The first entry in the following example, labeled ENTRY 7, is a panic frame. It shows a bus timeout error (a read timeout). Because the event type is PANIC, it is followed by a related log, labeled ENTRY 8, that records the error and status register values at the time of the error. See Appendix F for detailed information about memory registers.

The useful information in ENTRY 7 and ENTRY 8 is as follows:

- ENTRY 7 records the error.
- The OS EVENT TYPE field indicates the PANIC type.
- The PANIC MESSAGE field indicates that the error is a bus timeout.
- ENTRY 8 records the error and status register values at the time of the bus timeout error.
- The CAUSE register indicates a bus error during a data load or store operation.
- The BAD VIRT ADR register identifies the address of the timeout.

```

***** ENTRY 7. *****
----- EVENT INFORMATION -----
EVENT CLASS                ERROR EVENT
OS EVENT TYPE              200. PANIC
SEQUENCE NUMBER            2.
OPERATING SYSTEM           ULTRIX 32
OCCURRED/LOGGED ON         Tue Feb 26 06:22:14 1991 EST
OCCURRED ON SYSTEM         csselab1
SYSTEM ID                   x82030230  HW REV: x30
                             FW REV: x2
                             CPU TYPE: R2000A/R3000

PROCESSOR TYPE              KN02-BA
PANIC MESSAGE               bus timeout
***** ENTRY 8. *****
----- EVENT INFORMATION -----
EVENT CLASS                ERROR EVENT
OS EVENT TYPE              130. ERROR & STATUS REGS
SEQUENCE NUMBER            1.
OPERATING SYSTEM           ULTRIX 32
OCCURRED/LOGGED ON         Tue Feb 26 06:22:12 1991 EST
OCCURRED ON SYSTEM         csselab1
SYSTEM ID                   x82030230  HW REV: x30
                             FW REV: x2
                             CPU TYPE: R2000A/R3000

PROCESSOR TYPE              KN02-BA
----- KN02BA REGISTERS -----
CAUSE                       x0000081C
                             EXCEPTION CODE BUS ERR (DATA LOAD
                             _OR STORE)
                             HW INTERRUPT 1 PENDING

EPC                          x8010E164
STATUS                       x0000F204  CURRENT INTERRUPT STATE DISABLED
                             CURRENT MODE KERNEL
                             PREVIOUS INTERRUPT STATE ENABLED
                             PREVIOUS MODE KERNEL
                             OLD INTERRUPT STATE DISABLED
                             OLD MODE KERNEL
                             SW INTERRUPT 1 ENABLED
                             HW INTERRUPT 2 ENABLED
                             HW INTERRUPT 3 ENABLED
                             HW INTERRUPT 4 ENABLED
                             HW INTERRUPT 5 ENABLED
                             CACHE STATE NORMAL

BAD VIRT ADR                 xB0000000
SP                           xFFFFDDA8

```

SSR	x50070F00	LANCE DMA DISABLE SCSI DMA ENABLE SCSI DMA WRITE TO MEMORY COMM PORT 2 RECEIVE DMA DISABLE COMM PORT 2 TRANSMIT DMA ENABLE COMM PORT 1 RECEIVE DMA DISABLE COMM PORT 1 TRANSMIT DMA ENABLE
SIR	x00000002	
SIRM	xFF0F13E0	TOY INTERRUPT ENABLED SCC(0) SERIAL INTERRUPT ENABLED SCC(1) SERIAL INTERRUPT ENABLED ETHERNET INTERRUPT ENABLED SCSI INTERRUPT FROM 53C94 ENABLED LANCE DMA MEM READ ERR ENABLED SCSI DMA MEM READ ERR ENABLED SCSI DMA OVERRUN ERROR ENABLED SCSI DMA INTERRUPT ENABLED COMM PORT 2 RCV DMA OVERRUN ENABLED COMM PORT 2 RCV HALF PAGE INTR ENABLED

System Exercisers

The ULTRIX operating system contains a set of commands called exercisers. These exercisers reside in the /usr/field directory. They allow you to exercise the entire system or a specific subsystem.

The system exercisers test the following areas:

- System memory and file systems
- Peripherals
- Network and communications systems

Note: The ULTRIX subset UDTEXER must be installed for the exercisers to be present. To verify that UDTEXER is installed use the *setld -i* command.

On-line help

For further information on the ULTRIX system exercisers, there is a readmefirst file in the /usr/field directory. This file contains instructions on running all of the available exercisers. You can look at the file by typing

```
# more /usr/field/readmefirst
```

There are also help files for the exerciser. These help files can be accessed by using the -h option in the exerciser command. The following is an example of the help file for dskx:

```
# dskx -h
(dskx) - ULTRIX-32 Generic disk exerciser

Usage:
dskx [-h] [-ofile] [-t#] [-d#] -pdev#part or -cdev# or -rdev#
-h          (optional) Print this help message.
-ofile      (optional) Save output diagnostics in file.
-d#         (optional) Print statistics every # minutes.
-t#         (optional) Run time in minutes (default: run until
                  cntl-c).
-pdev#part  Exercise partition only, write/read.
-rdev#      Exercise random partitions, read only
-cdev#      Exercise random partitions, write/read.

dev# - Device name raw or buffered and number
      (i.e. rhp0, ra3, rkl).
part - partition [a - h]

Examples:
      dskx -t60 -cra1 &
(Exercise RA type disk, number 1, for 60 min. in background)
      dskx -prhp0d
(Exercise raw HP type disk, number 0, partition d, run forever)
```

Note: *The ULTRIX Guide to System Exercisers contains additional reference material.*

Running the system exercisers

To run any exerciser you must be logged in as superuser and /usr/field must be your current directory. Type the following:

```
% su
password:
# cd /usr/field
```

You can run the exercisers two ways; you can run a script containing several exercisers or you can run a specific exerciser individually.

Running several exercisers at once

To run more than one exerciser at the same time, use the syscript command.

```
# syscript
```

Syscript will prompt you for the exercisers you wish to run, and for how long you wish to run them. The syscript command then executes the individual exercisers and creates a file called testsuite. You can reexecute the commands in testsuite by entering:

```
# sh testsuite
```

Running individual exercisers

Individual exercisers are grouped into three categories:

- File systems and memory
- Peripherals
- Network and communications systems

Exercising the file system and memory

This section describes how to test local file systems, system memory, and shared memory.

Exercising local file systems

Use the fsx command to exercise the file systems locally. The fsx command exercises the specified file system by initiating multiple processes, each of which creates, writes, closes, opens, reads, validates, and unlinks a test file of random data. The format of the fsx command is:

```
fsx -h -ofile -pn -fpath -tmin
```

-h Prints the help message for the fsx command.

-ofile Saves the output diagnostics in file.

-pn Specifies the number(n) of processes you want fsx to initiate. The maximum is 250, the default is 20.

-fpath Specifies the pathname of the directory of the file system to test. The default is /usr/field.

-tmin Specifies how many minutes you want fsx to run.

The following is an example of the fsx command:

```
# fsx -p5 -f/usr -t60 &
```

This example starts 5 processes and tests the /usr file system for 60 minutes in the background.

Exercising the system memory

Use the memx command to exercise system memory. The memx command exercises the system memory by initiating multiple processes. The memx command runs ones and zeros, zeros and ones, and random data patterns in the allocated memory being tested. The format of the memx command is:

```
memx -h -ofile -s -mn -px -tmin
```

-h Prints the help message for the memx command.

-ofile Saves the output diagnostics in file.

-s Disables automatic invocation of shm.

-mn Specifies the amount of memory in bytes to test. The default is total memory divided by 20.

-px Specifies the number of processes to initiate. The maximum allowed is 20 which is also the default.

-tmin Specifies how many minutes you want memx to run.

The following is an example of the memx command:

```
# memx -s -m4095 -p5 -t60 &
```

This example disables the shared memory exerciser, tests 4095 bytes of memory, starts 5 processes and runs for 60 minutes in the background.

Exercising shared memory

Use shmx to exercise shared memory. The shmx command spawns a background process called shmxb, and together shmx and shmxb exercise the shared memory segments. They take turns writing and reading each other's data in the segments. The format of the shmx command is:

```
shmx -h -ofile -ti -mj -sk -v
```

- h Prints the help message for the shmx command.
- ofile Saves diagnostic output in file.
- ti Indicates the run time in minutes (i).
- mj Specifies the memory segment size in bytes (j) to be tested.
- sk Specifies the number of memory segments (k). The maximum and default is 6.
- v Uses the fork system call instead of the vfork system call to spawn shmxb.

The following is an example of the shmx command:

```
# shmx -t180 -m100000 -s3 &
```

This example runs for 180 minutes, tests 100,000 bytes on three memory segments, and runs in the background.

Exercising peripherals

This sections describes the system exercisers you use to test disk drives, tape drives, and line printers.

Exercising disk drives

Use `dskx` to exercise disk drives. The `dskx` command exercises specified partitions and file systems on the designated disk. The format of the `dskx` command is:

Note: The `-p` and `-c` options destroy data on the device you are testing. Use extreme caution when using either of these options.

```
dskx -h -ofile -pdevpart -cdev -rdev -tmin -dn
```

<code>-h</code>	Prints the help message for the <code>dskx</code> command.
<code>-ofile</code>	Saves output diagnostics in file.
<code>-pdevpart</code>	Performs random seeks, writes and reads on the specified partition (part) of the device (dev). Next it validates the random data and block sizes. You cannot test the c partition because the test would corrupt the bad block information.
<code>-cdev</code>	Performs random seeks, writes, and reads on all partitions of the specified device (dev) except the c partition. Partition c is not tested because the test would corrupt the bad block information.
<code>-rdev</code>	Performs random seeks and reads on all partitions of the device (dev) except partition c. The <code>-r</code> option is safe to use on any disk because it will not overwrite data.
<code>-tmin</code>	Indicates the run time in minutes.
<code>-dn</code>	Specifies, in minutes, how often you want the <code>dskx</code> command to print diagnostics to the terminal. The default is to print diagnostics upon completion of the exercise.

The following is an example of the `dskx` command:

```
# dskx -rrz0 -t20 -d5 &
```

This example runs on `rz0` for 20 minutes, and diagnostic information is displayed on the terminal every 5 minutes. This example also runs in the background.

Exercising tape drives

There are two tape drive exercisers, `mtx` and `tapex`. Both exercisers write, read, and validate random data on a tape device from the BOT to EOT. `Tapex` performs additional tests, for example, positioning tests for records and files.

The `mtx` tape drive exerciser

The format of the `mtx` command is:

```
mtx -h -ofile -rn -fn -sdev# -ldev# -vdev# -adev# -tmin
-h          Prints help message for the mtx command.
-ofile      Saves diagnostic output in file.
-rn         Specifies record length in bytes.  Default
            value is 10240.
-fn         Specifies the length of the file in numbers
            of records.
-sdev#      Performs a short record test that writes,
            reads, and validates 512-byte records on
            device (dev). The dev# variable is the raw
            device name and number, such as rmt0h.
-ldev#      Performs a long record test that writes,
            reads, and validates 10240-byte records on
            device (dev). The dev# variable is the raw
            device name and number, such as rmt0h.
-vdev#      Performs a variable record length test
            that writes, reads, and validates random
            record lengths from 512 to 20280 bytes
            on device (dev). The dev# variable is the
            raw device name and number, such as rmt0h.
-adev#      Performs short, long, and variable record
            length tests on device (dev). The dev#
            variable is the raw device name and number,
            such as rmt0h.
-tmin       Specifies how many minutes you want mtx to
            test the tape drive.
```

The following is an example of the `mtx` command:

```
# mtx -r20480 -lrmt0h -t60 &
```

This example writes 20,480 byte records to rmt0h for 60 minutes and runs in the background.

The tapex tape drive exerciser

The format of the tapex command is:

```
tapex -options -parameters
```

Some tapex options cause specific tests to be performed, for example, an end-of-media test. Other options modify the tests, for example, by enabling caching. The tapex options are as follows:

Like all ULTRIX commands the tapex options are case sensitive:

- a Performance measurement test.
- b Continuous read/write test.
- c Enables caching on the device.
- C Disables caching on TMSCP devices.
- d Tests ability to append the media.
- e End-of-media test.
- E Runs an extensive series of tests in sequential order.
- f Specifies the name of the device special file that corresponds to the tape unit being tested.
- F File-positioning tests.
- G File-positioning tests on already-written tape.
- g Random record-size tests.
- h Display a help message describing tapex.
- i Interactive mode.
- j Write phase of the tape-transportability tests.
- k Read phase of the tape-transportability tests.
- L Media loader test.
- l End-of-file test.
- m Displays tape contents.
- N Disables the usage of n-buffered I/O on tests that support it's usage.

- o Sends output to the specified file name.
- p Runs both the record-positioning and file-positioning tests.
- q Command timeout test.
- r Record-size test.
- R Record-positioning test.
- s Record-size behavior test.
- S Single record size test.
- T Copies output to standard output.
- v Verbose mode.
- V Very verbose mode.
- w Opens the tape as read-only.
- Z Initializes the read buffer to the nonzero value of 0130.

The following is an example of the tapex command:

```
# tapex -f /dev/rmt0h -E -o tapex.out
```

This example runs an extensive test in sequential order on /dev/rmt0h and sends the results to tapex.out.

Exercising the network and communication systems

This section describes how to test the following network and communication systems:

- TCP/IP
- Terminal communications

Exercising the TCP/IP network

Use the netx command to exercise the TCP/IP network. The netx command sets up a stream socket connection with netx acting as the client and the miscd utility acting as the server in the TCP/IP internet domain. Using the connection, netx writes random data to the miscd server. The server loops the data back to netx, and then the data is read and verified against the original data. The format of the netx command is:


```

netx -h -pn nodename -tmin
-h          Prints the help message for the netx command.
-pn         Specifies the port number to use in the
            internet domain. The variable n must be
            less than 32768.
nodename    The name of the remote or local system host
            running the server.
-tmin       Specifies how many minutes you want netx
            to run.

```

The following is an example of the netx command:

```
# netx tinker -t60 &
```

This example runs a test on node tinker for 60 minutes and is running in the background.

Exercising the terminal communications

Use the cmx command to exercise the terminal communications system. The cmx command writes, reads, and validates random data and packet lengths on the communications line or lines specified. The format of the cmx command is:

```

cmx -h -ofile -tmin -l line-1 line-2 line-n...
-h          Prints help message for the cmx command.
-ofile      Saves output diagnostics in file.
-tmin       Specifies how many minutes you want cmx to run.
-l line...  Specifies the line or lines you want to test.

```

Note: *You must have a loopback connector on the line you wish to test.*

The following is an example of the cmx command:

```
# cmx -l 00 -t45 &
```

This example runs a test on tty00 for 45 minutes and runs in the background.

Exercising line printers

Use the `lpx` command to exercise line printers. The `lpx` command exercises the line printers by printing a rolling character pattern repeatedly to the device. The format of the `lpx` command is:

```
lpx -h -ofile -pn -ddev -tmin
-h          Prints the help message for the lpx command.
-ofile      Saves diagnostics output to file.
-pn         Specifies the pause period in n minutes.
-ddev       Specifies the line printer you want to test.
-tmin       Specifies how many minutes you want lpx
            to exercise the line printer.
```

The following is an example of the `lpx` command:

```
# lpx -t60 -dlp1 &
```

This example runs a test on `lp1` for 60 minutes; it runs in the background.

Obtaining a Crash Dump

This chapter discusses the following topics:

- Verifying the system is hung
- Crashing the system
- Creating a dump file
- Setting the haltaction Environmental Variables

Troubleshooting Techniques

This chapter discusses how to force a crash dump on a hung system. This enables you to capture the state information contained in memory. A dump provides a snapshot of memory just prior to the system crashing or hanging. The dump can be used to reconstruct what transpired in the system during this period of time.

It is important that the contents of memory be saved so that they are available for analysis if needed. The state information is useful in determining why a system crashed or hung. Once the system is rebooted all state information is lost.

Analyzing a crash dump requires an in-depth knowledge of both the operating system and the hardware. This is usually done by a support engineer. Analysis of the dump is not covered in this manual.

The material presented in this chapter assumes the reader has a working knowledge of the ULTRIX operating system.

Verifying the system is hung

A wedged software application can exhibit the same symptoms as a hung system, so it is important to determine whether the software applications are hung or whether the entire system is hung. One good way to make this determination is to successfully perform any of the tasks listed below:

- Pop up another window.
- Log in from an attached terminal (if one is connected)
- **rlogin** or **dlogin** from another system on the network.
- "Ping" the system from the network (use the ULTRIX *ping* command described in the man pages).

If any of these tasks complete successfully, you have demonstrated that the system can execute commands. In this case contact the system manager and have the software application restarted.

If these tasks are unsuccessful in executing, it is a good indication that the system is not capable of executing instructions and is in a hung state. In this case you should crash the system and create a dump file.

Crashing the system

Before placing the contents of memory in a dump file, enter the hardware console mode by abruptly halting the system. The system will then transfer control of the processor from the operating system to the ROM based console program. The console program runs out of ROM, not system memory. This enables the CPU to process instructions while not modifying contents of system memory.

To force the system to halt, simultaneously depress **Ctrl**, **Alt** and **Return** keys on the LK501 key board attached to the SERIAL.bus. The system will then display the >> prompt.

There are times when it will not be possible to halt the system. If any of the system hardware is in a hung condition, then the halt directive will not be processed.

When the **Ctrl**, **Alt** and **Return** keys are depressed, a halt packet is sent over the SERIAL.bus to the dedicated SERIAL.bus controller chip on the system module. This dedicated processor in turn toggles an interrupt line that attaches directly to the CPU. The CPU then services the interrupt and jumps to an address that executes the console program stored in ROM.

Creating a dump file

Once in console mode a dump file can be generated by issuing the **>>go 0x80030008** command. It will take approximately 2 minutes to generate the dump file.

The **>>go** command causes the system to execute a program located at address **0x80030008** in the portion of memory that contains the kernel. The dump file will be placed in the file system you specified when the kernel was built.

The default dump file's path specification is */usr/adm/crash*. The file has the potential to be as large as the physical memory installed in the machine. However, the routines that create the dump have been optimized to only capture the significant portions of memory. In practice, the dump file is significantly smaller than that of physical memory.

***Note:** It is a good practice to run the thorough power up self-test script with the **>>pst-t** command before rebooting the system. This will help insure that a hardware problem did not cause the system to hang.*

Setting the haltaction Environmental Variables

When troubleshooting a system with a chronic hang or crash problem, it is good practice to set the **haltaction** environmental variable to **h** by issuing the **>>set haltaction h** command.

This will prevent the system from automatically rebooting itself, thereby preserving the contents of memory.

A machine who's haltaction is set to **r** or **b** will try to boot or run the powerup self-test after it crashes. To stop this from happening and preserve memory quickly type **Ctrl-C**. This will cause the console's boot and self-test programs to abort. You will then be returned to the **>>** prompt.

Procedural Flow Charts

This chapter contains the following procedural flow charts:

- Troubleshooting when the monitor has no display
- Troubleshooting a missing SCSI device
- Troubleshooting a missing TURBOchannel option
- Troubleshooting missing memory SIMMs
- Troubleshooting a missing SERIAL.bus device

This chapter documents troubleshooting procedures that are used when a system has no display on the video console, and when specific components installed in the system do not show up in the configuration.

No Display at Power-Up

When a system has no video console display you cannot run diagnostics and interpret error codes. In these instances it is necessary to follow the steps outlined in Figure 9-1.

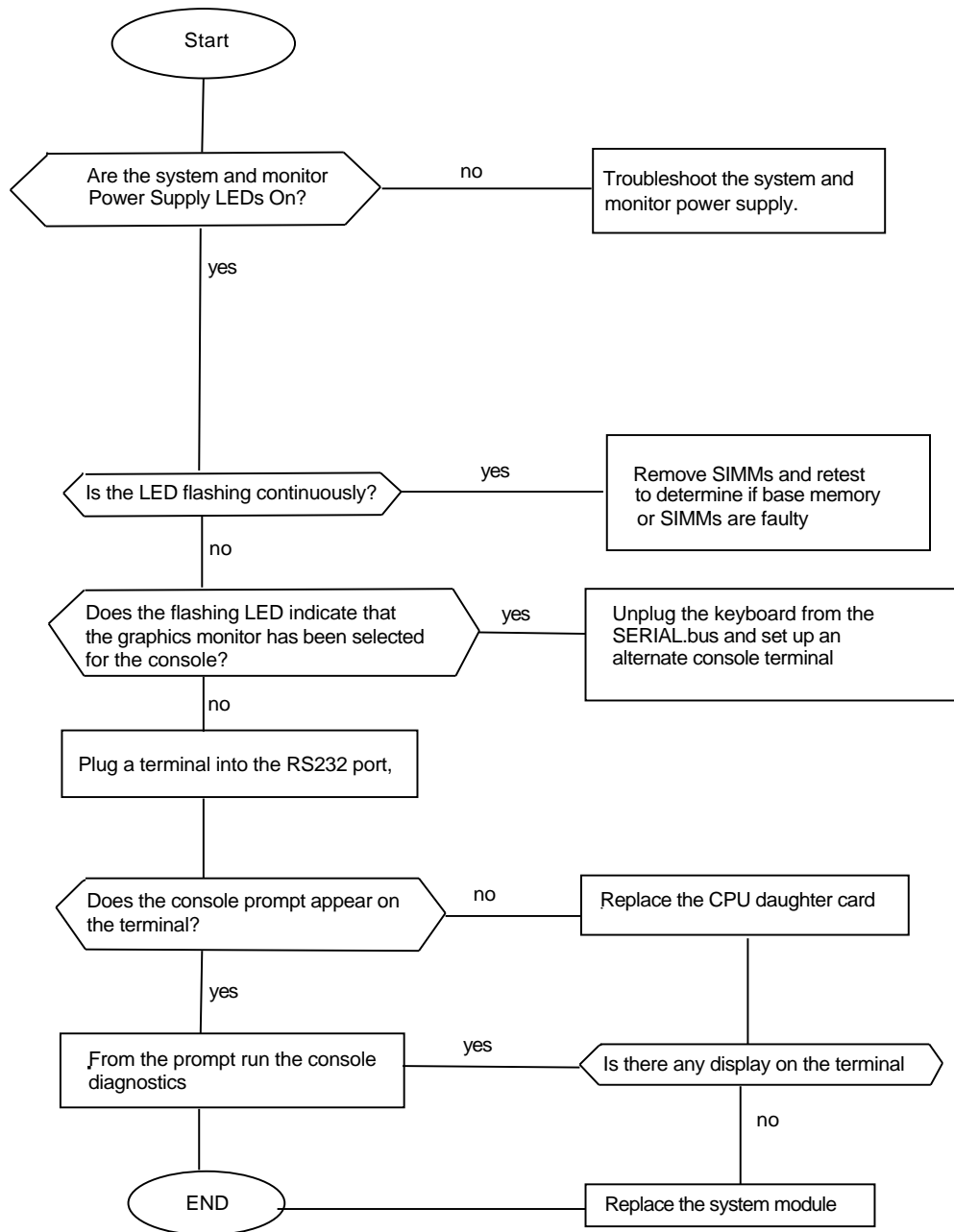


Figure 9-1. Troubleshooting When the Monitor Has No Display

Missing SCSI Device

Frequently, problems occur on the SCSI bus that cannot be isolated with diagnostics. Drives must appear in the configuration before they can be accessed and tested. Figure 9-2 outlines the steps used to troubleshoot this class of SCSI bus problem.

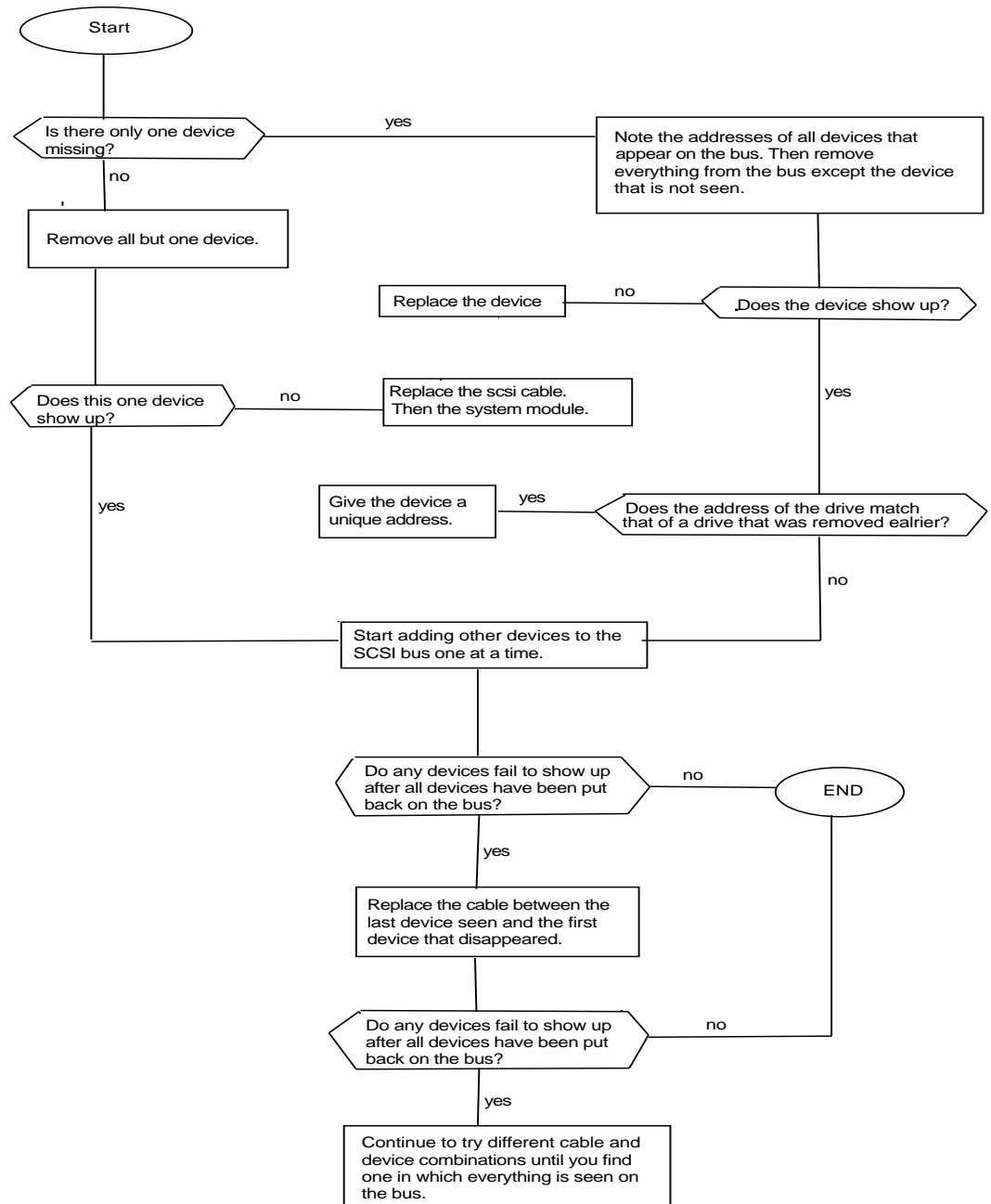


Figure 9-2. Troubleshooting a Missing SCSI Device

Missing TURBOchannel Option

TURBOchannel options, like SCSI modules, must appear in the configuration before they can be accessed and tested. Figure 9-3 explains how to troubleshoot a missing TURBOchannel option.

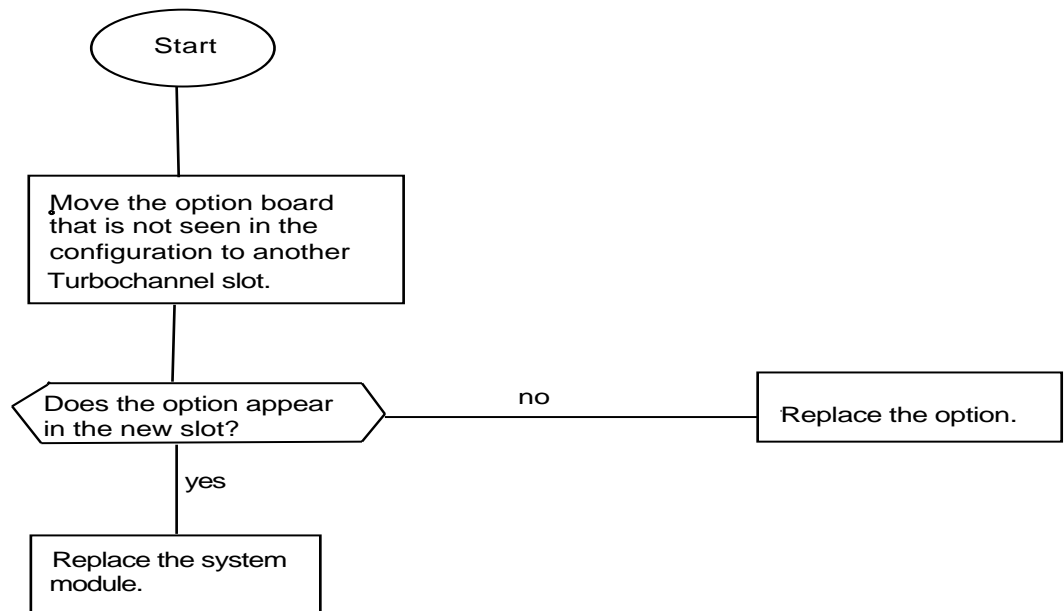


Figure 9-3. Troubleshooting a Missing TURBOchannel Option

Missing Memory SIMMs

There are times when memory physically installed in the system does not appear in the configuration. This can be caused by improper seating or by a faulty memory module. Such problems can degrade system performance. Figure 9-4 shows how to determine which memory SIMMs are faulty.

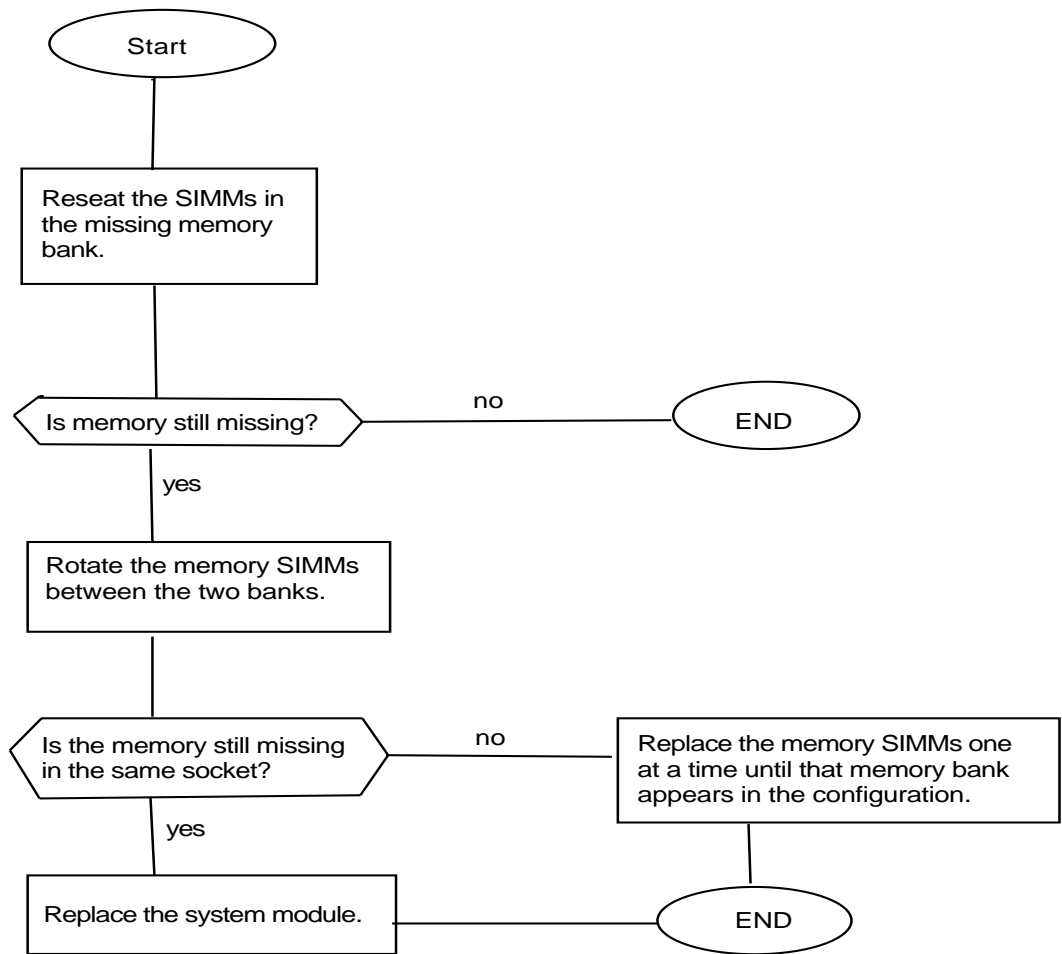


Figure 9-4. Troubleshooting Missing Memory SIMMs

Missing SERIAL.bus Device

A faulty or missing SERIAL.bus keyboard can prevent the system from displaying messages on the video console. Before you open the system and replace any modules, follow the steps outlined in Figure 9-5.

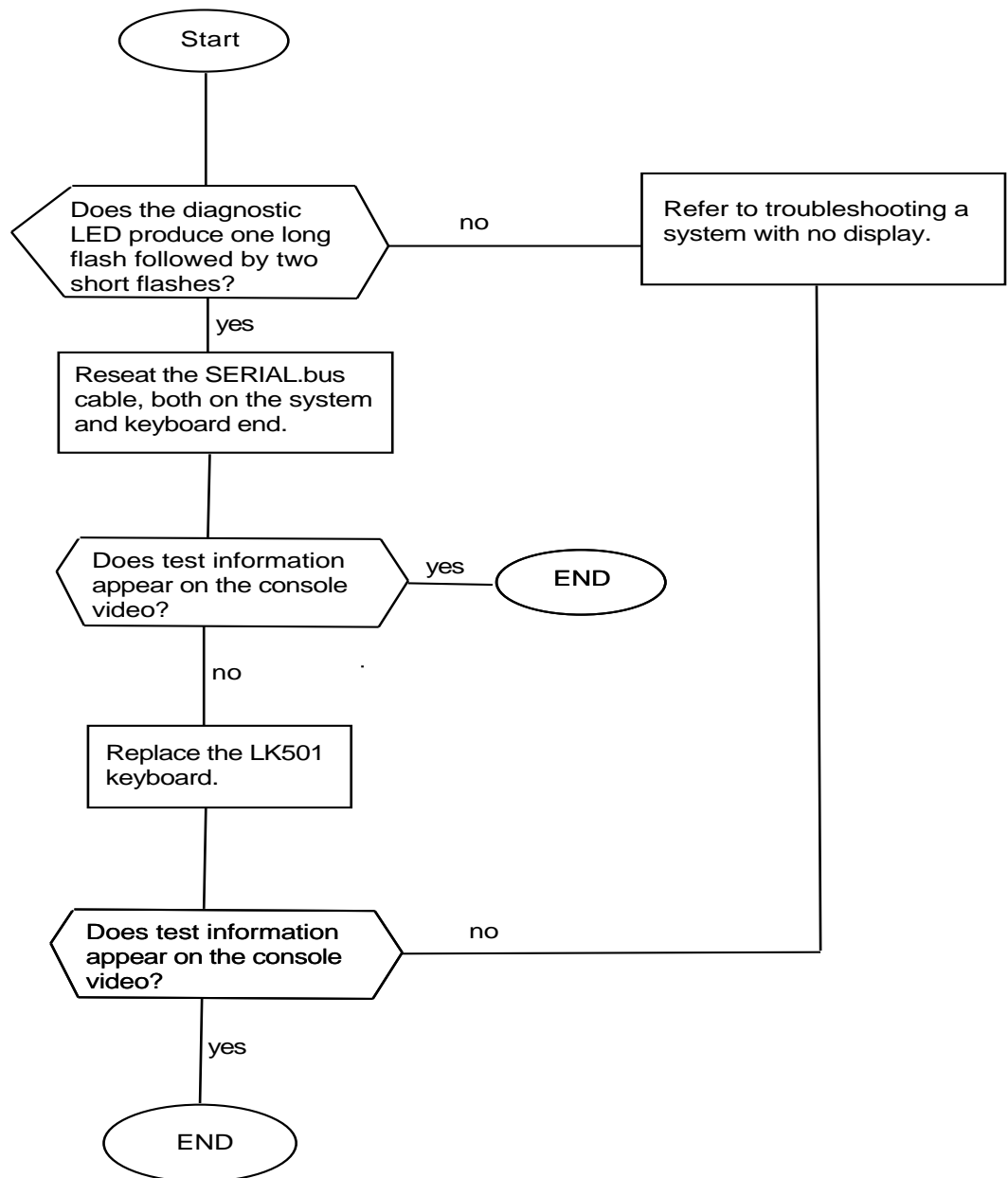


Figure 9-5. Troubleshooting a Missing SERIAL.bus Device

Part III

Removing and Replacing FRUs

Base System Hardware

This chapter describes

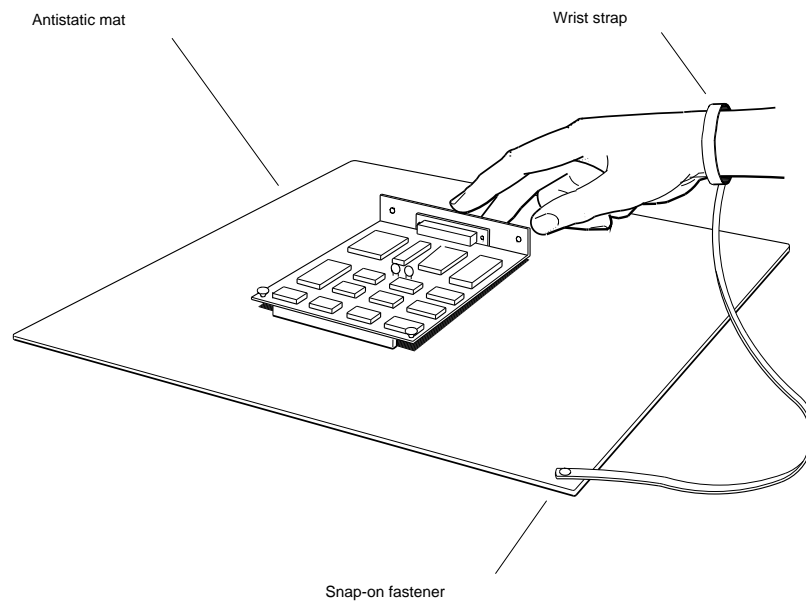
- How to avoid electrostatic damage when handling components
- How to remove and install the system unit cover
- How to install and remove terminators and loopback connectors

Using the Antistatic Kit

The Digital antistatic kit contains the equipment needed to handle static-sensitive electronic components safely.

Use the antistatic kit and handle static-sensitive components as follows:

1. Connect the antistatic mat to the system chassis.
2. Place the antistatic wrist strap around your wrist and attach the other end of the strap to the snap-on fastener on the antistatic mat.
3. Place static-sensitive components on the antistatic mat when you put them down.



WSMXM009

Figure 10-1. Using the antistatic kit

Removing and Installing the System Unit Cover

To Remove the System Unit Cover

1. Remove the two screws that fasten the system unit cover to the rear panel of the system unit chassis, as shown in Figure 10-2.

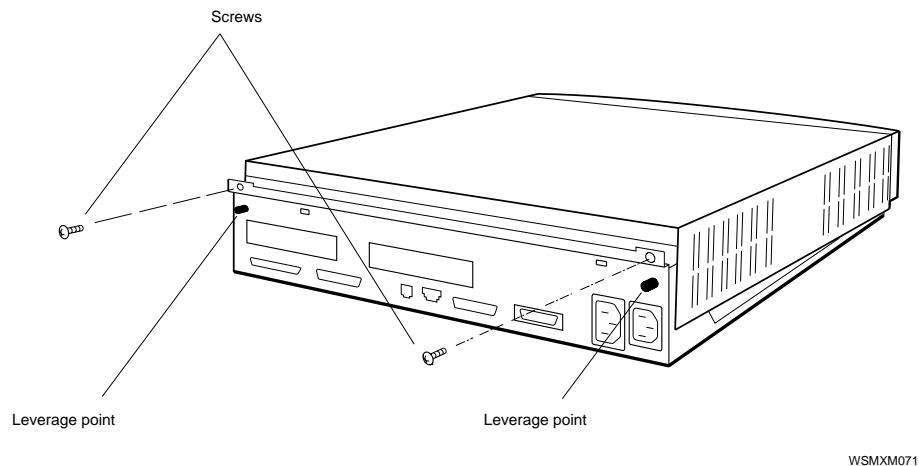


Figure 10-2. System unit cover

2. Pressing down on the two black leverage points on the back of the system unit, and at the same time pressing up against the edge of the cover, lift the back end of the cover about 6 inches.

If you need to exert more pressure to lift the cover, use the system tool provided for that purpose.

Insert the tool into one of the two small holes just beneath the edge of the cover on the back of the system unit. Lift up on the tool until the cover loosens. Insert the tool into the other hole and repeat the procedure.

3. When the tabs are loose in their slots in the front of the cover, lift the cover up and away from the system unit.

Note: Do not stand the cover on its face. The paint is easily scratched.

To Install the System Unit Cover

1. Make sure all cables are safely within the enclosure before installing the cover. Cables can be easily scratched if caught by the cover.
2. Tilt the cover so the back is about 15 centimeters (6 inches) higher than the front.
3. Insert the tabs on the front edge of the cover into the slots that run along the front edge of the system unit.
4. Lower the cover into place and replace the screws that hold it.

Installing and Removing a Serial Line Adapter

Use the serial line adapter (12-33190-01) to connect a modified modular jack (MMJ) cable connector from a serial device to the D-connector of the system's RS-232 serial communications port. The serial line adapter is also called the comm-line-to-MMJ adapter.

To install a serial line adapter

Press the keyed serial line adapter onto the serial connector and tighten the two screws.

To remove a serial line adapter

Loosen the two screws that hold the adapter to the serial connector, and pull the adapter off of the serial connector.

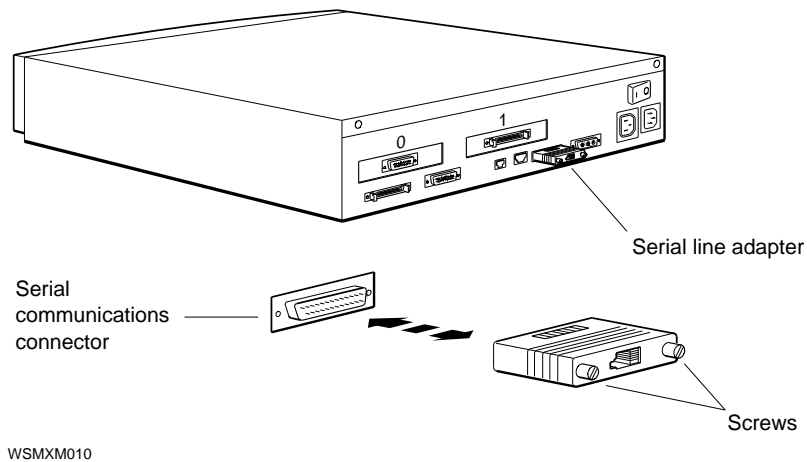


Figure 10-3. Serial line adapter

Installing and Removing a SCSI Controller Terminator

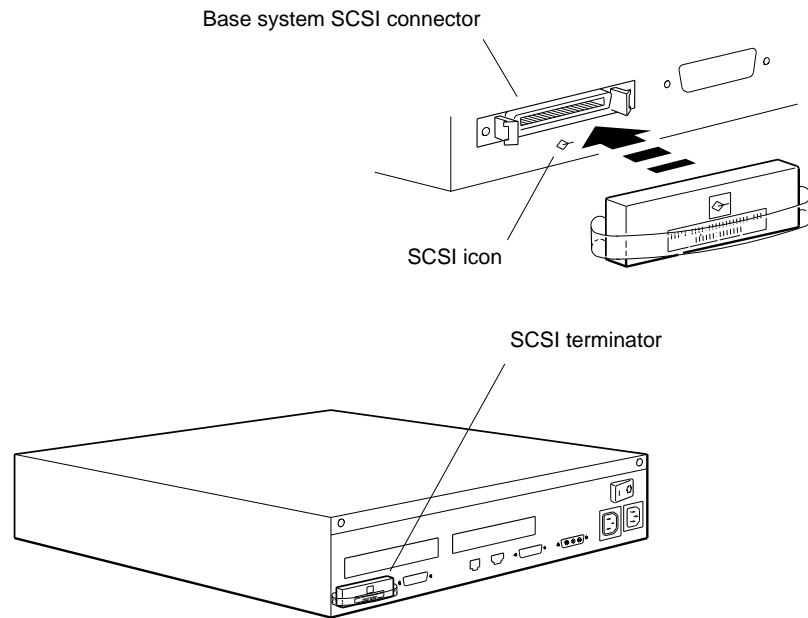
A SCSI controller terminator (12-33626-01) must be installed on the connector of any SCSI controller that has no external device connected to it.

To Install a SCSI Controller Terminator

Firmly press the keyed terminator onto the SCSI connector.

To Remove a SCSI Controller Terminator

Grip the loop on the SCSI controller terminator and pull the terminator off of the SCSI connector.



WSMXM011

Figure 10-4. SCSI controller terminator

Installing and Removing a SCSI Chain Terminator

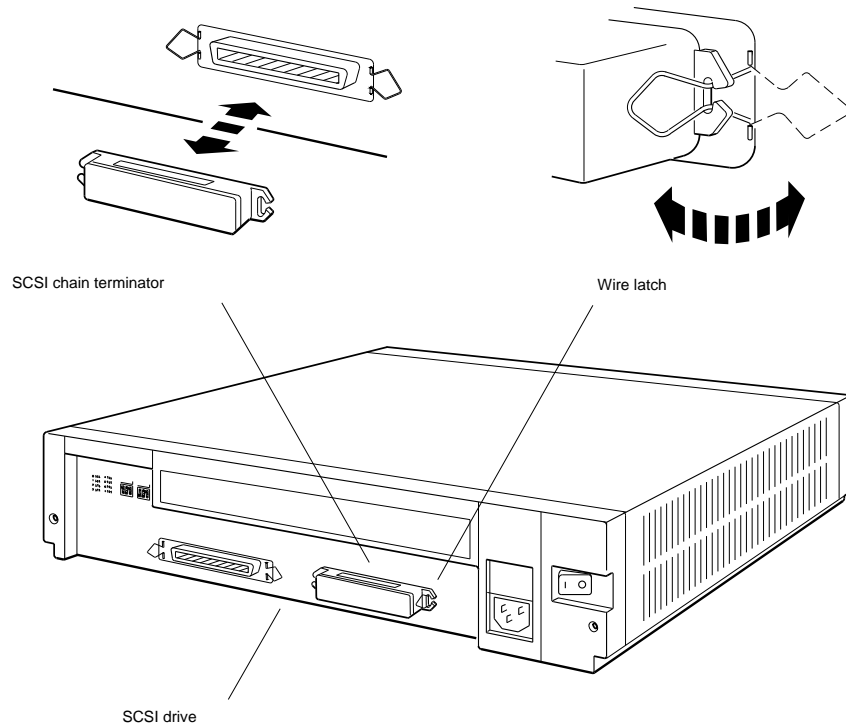
The last SCSI drive in a chain requires a SCSI chain terminator (12-30552-01) on its output (right) connector.

To Install a SCSI Chain Terminator

1. Firmly press the daisy chain terminator onto the unused connector on the drive at the end of the chain. It fits when the widest part of the connector is on top.
2. Press the wire latches on the SCSI connector onto the SCSI terminator.

To Remove a SCSI Chain Terminator

Push the wire latches away from the terminator and pull it off of the SCSI connector.



WSMXM012

Figure 10-5. SCSI chain terminator

Installing and Removing a ThickWire Ethernet Loopback Connector

The ThickWire Ethernet loopback connector (12-22196-02) must be installed for certain Ethernet controller (base system or TURBOchannel option) tests.

To Install a ThickWire Ethernet Loopback Connector

Firmly press the keyed loopback connector onto the ThickWire Ethernet connector. Press the sliding lock on the system unit connector toward the connector to hold the terminator firmly in place.

To Remove a ThickWire Ethernet Loopback Connector

Unlock the sliding lock then firmly grip the ThickWire loopback connector and pull it away from the ThickWire connector.

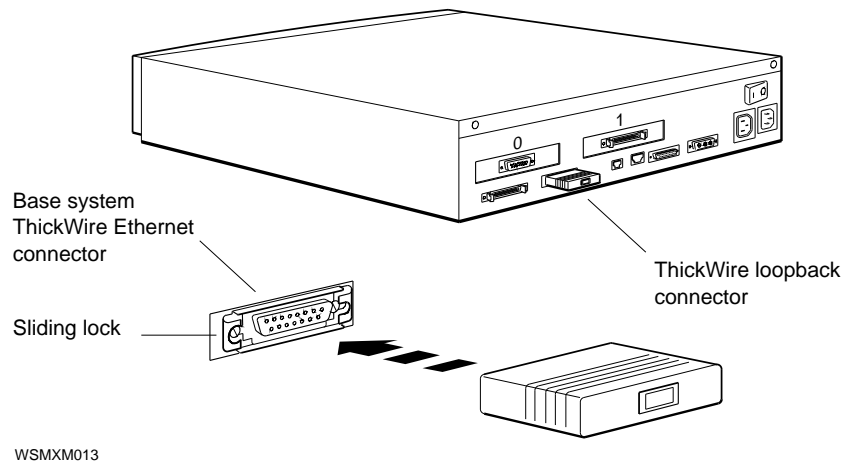


Figure 10-6. ThickWire Ethernet loopback connector

Installing and Removing a Serial Line Loopback Connector

The serial line loopback connector (12-25083-01) must be installed if you want to perform certain serial controller tests. The serial line loopback connector is also called the modified modular jack (MMJ) loopback connector.

To Install a Serial Line Loopback Connector

Firmly press the keyed connector onto the serial line connector.

To Remove a Serial Line Loopback Connector

Firmly grip the serial line loopback connector and pull it away from the serial line connector.

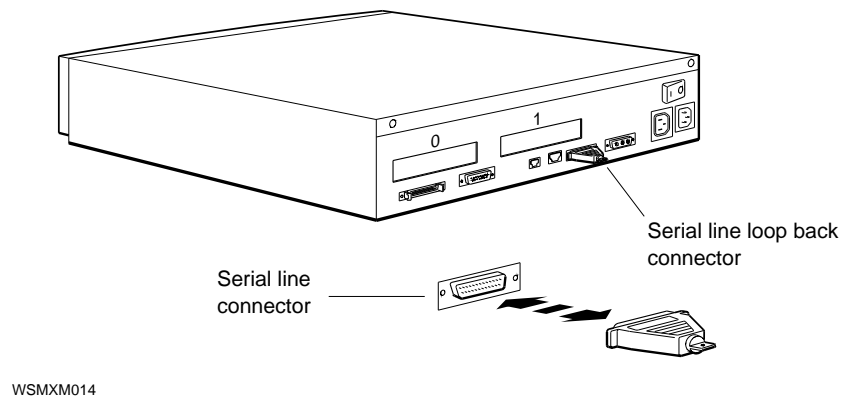


Figure 10-7. Serial line loopback connector

Storage Devices

This chapter discusses the storage drives used with the Personal DECstation/DECsystem. It covers the following topics:

- SCSI requirements for storage devices
- Which internal drives can be installed in the Personal DECstation/DECsystem enclosure
- How to remove and install internal drives and associated hardware
- How to set the SCSI ID on internally mounted drives

SCSI Requirements

For the system to work properly, SCSI drives must be connected to base system or TURBOchannel option SCSI controllers according to the following guidelines.

- The base system SCSI controller has a built-in terminator. It is not necessary to install a terminator on the internal SCSI connector.
- If no external drives are installed, the external connector on any SCSI controller must be terminated with a SCSI controller terminator (12-33626-01).
- A SCSI controller can accommodate up to seven drives linked together in a daisy chain.
 - On the base system SCSI controller, the optional internal hard disk drive counts as the first drive. Six more external drives can be connected to that controller.
 - If there is no internal drive, seven external drives can be connected to that controller.
 - TURBOchannel option modules can support up to seven external drives in a daisy chain attached to each SCSI option module connector on the back of the system unit.
- Each drive enclosure has two SCSI connectors, except for the TURBOchannel extender (TCE) box, which has only one.
- The second connector of the last drive in a chain, must contain the SCSI daisy chain terminator that came with the drive. The TCE box has its own internal terminator and must always be the last box in a chain.
- Each drive in a daisy chain must have a unique SCSI ID from 0 to 6.
- The SCSI controller always has SCSI ID 7.
- The length of an internal and external cables connecting the drives in any one chain must not exceed 6 meters (236 inches). See Table 11-1.

- As a rule of thumb, a SCSI chain should consist of no more than three external expansion boxes supplied by Digital, regardless of the number of drives each box contains. Additionally, the overall total of seven drives per SCSI connector must not be exceeded.

SCSI Cable Lengths

Use the cable lengths in Table 11-1 to compute the total SCSI cable length for any combination of expansion boxes supplied by Digital.

The combined length of all internal and external cables connecting the drives in a given SCSI chain must not exceed 6 meters (236 inches).

Table 11-1. SCSI Cable Lengths

Cable or box	Cable length	
	cm	in
Base system SCSI controller and internal cable	40	16
BA40 storage expansion box internal cable	36	14
BA42 storage expansion box internal cable	79	31
RRD40 optical compact disk drive internal cable	7	3
TLZ04 tape drive internal cable	97	38
50-pin to 50-pin box-to-box external cable	66	26
50-pin to 50-pin system-unit-to-expansion-box external cable	96	38

Internal Drives

Two drives can be present within the Personal DECstation /DECsystem unit enclosure.

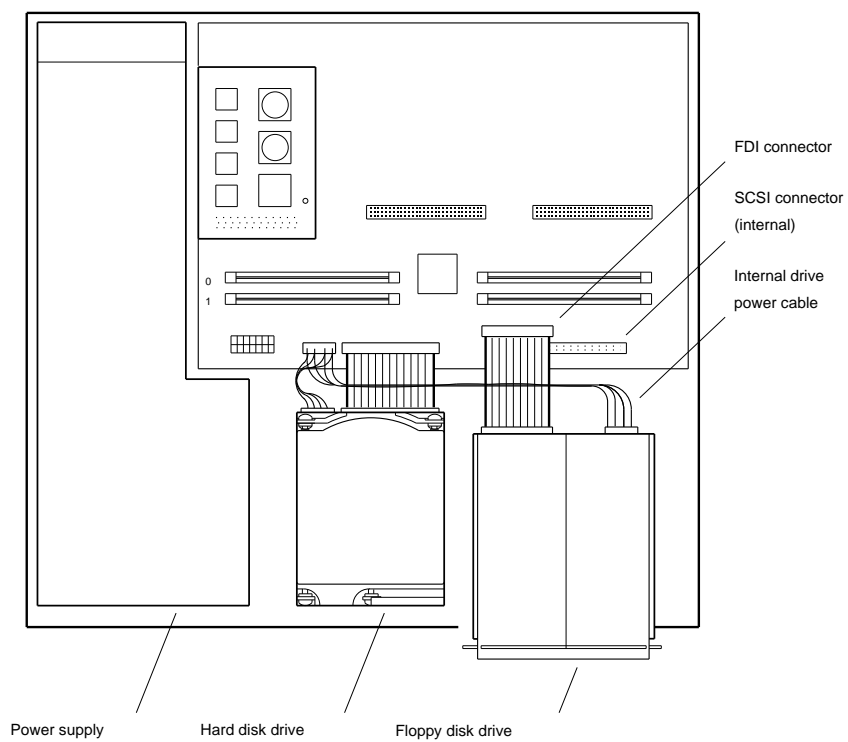
- One 2.88-megabyte RX26 floppy disk drive can be installed in the floppy disk mounting bracket in the right front corner of the enclosure.

Insert and remove floppy disks in the drive through the opening in the front panel of the system unit.

Note: *The internal floppy disk drive is not a SCSI device.*

- One 426-megabyte RZ25 hard disk drive can be installed on the floor of the system unit chassis between the floppy disk mounting bracket and the power supply. The internal hard disk drive is a SCSI device.

For information about troubleshooting the internal drives, see Chapters 4, 5, 6, 7, and 9 of this guide.



WSMXM029

Figure 11-1. Internal drives

Removing and Installing an Internal Floppy Disk Drive

One RX26 floppy disk drive (RX26-AA) can be installed in the floppy disk mounting bracket in the right front corner of the enclosure as shown in Figure 11-3. The FDI cable (17-02221-02) connects the floppy disk drive to the FDI connector on the base system module. The internal drive power cable (17-03357-01) conducts DC power to the floppy disk drive from the internal drive power connector on the base system module.

To remove a floppy disk drive

1. Release the sliding latch that fastens the floppy disk mounting bracket to the system unit chassis. See Figure 11-2.

This enables the bracket to move around for access to the connectors and to the screws that hold the drive in the mounting bracket. See Figure 11-3.
2. Disconnect the FDI and power cables from the floppy disk drive and from the base system module.
3. Remove the four screws that fasten the shield and drive assembly in the mounting bracket. Remove the shield and drive assembly from the bracket.
4. Slide the drive out of the front of the shield.

To install a floppy disk drive

1. If necessary, remove the four screws that hold the shield in the floppy disk mounting bracket. Remove the shield from the bracket.
2. Position the floppy disk drive in front of the shield. The top of the drive is a plain metal surface, and the bottom is a printed circuit board. The top of the shield is a plain metal surface, and the bottom is a perforated metal surface. The front of the shield is flanged, and the rear is a simple metal edge.
3. Slide the drive into the shield. Make sure the four screw holes in the shield are aligned with the four screw holes in the drive.

4. Release the sliding latch that fastens the floppy disk mounting bracket to the system unit chassis. This enables the bracket to move around for access when installing the connectors and screws.
5. Place the shield and drive assembly in the bracket and fasten it in place with the four screws.
6. Connect the FDI cable to the drive and to the FDI connector on the base system module. Connect the internal drive power cable to the drive and to the drive power connector on the base system module.
7. Refasten the sliding latch that fastens the floppy disk mounting bracket to the floor of the system unit chassis.

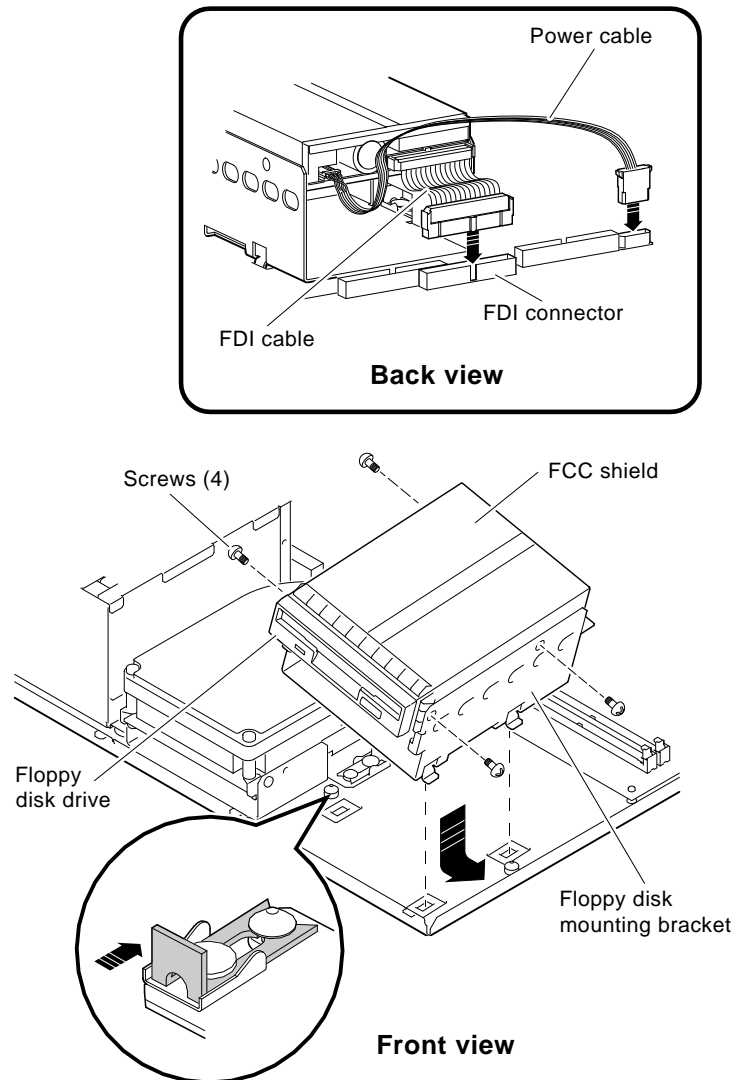


Figure 11-2. Floppy disk drive

LJ-02976-T10

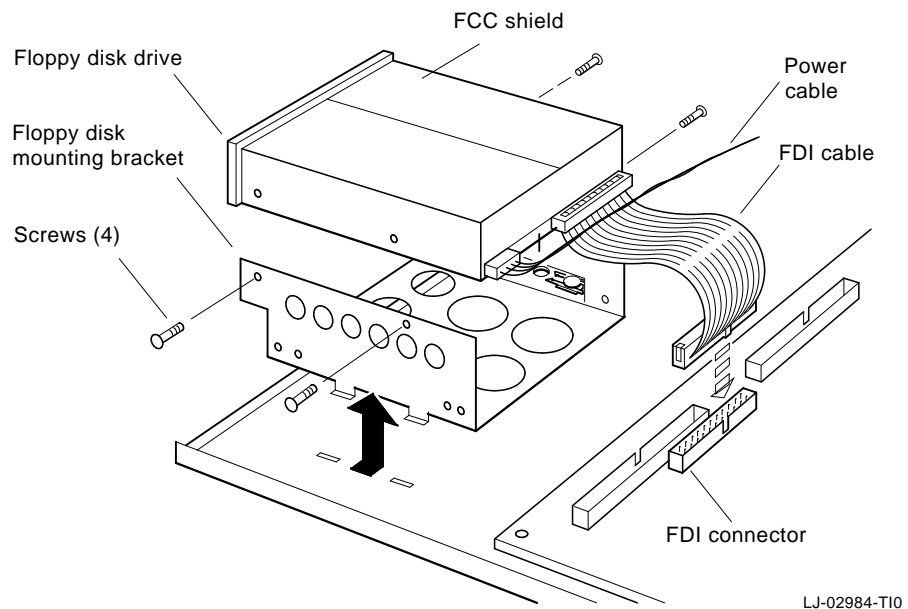


Figure 11-3. Floppy disk drive

Removing and Installing a Floppy Disk Mounting Bracket

The floppy disk mounting bracket assembly (70-28359-02) holds the floppy disk drive in place within the system unit enclosure. The bracket is mounted near the front right corner of the system unit chassis. It is held in place by two tabs that are inserted into slots in the system unit chassis and one sliding latch that locks the bracket into place.

To remove a floppy disk mounting bracket

1. Release the sliding latch that fastens the bracket assembly to the floor of the system unit chassis.
2. If a floppy disk drive is in the floppy disk mounting bracket, disconnect the cables from the drive.
3. Tilt the bracket assembly away from the power supply, allowing it to pivot around the tabs in the slots. When the tabs are free, lift the bracket assembly up and out of the system unit chassis.

To install a floppy disk mounting bracket

1. Position the floppy disk drive mounting bracket over the system unit chassis with the two tabs pointed down into the slots in the chassis. Slide the tabs into the slots.
2. Lower the bracket assembly to the floor of the chassis.
3. Engage the sliding latch to fasten the bracket assembly in place.

Removing and Installing a Floppy Port Cover

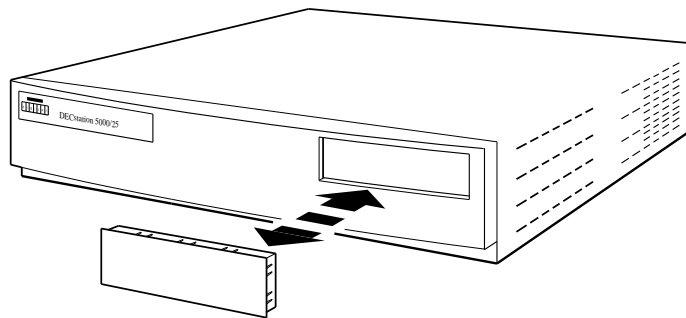
The floppy port cover (74-46387-01) fills the opening in the front panel of the system unit cover when no internal floppy disk drive is installed.

To remove the floppy port cover

Reach inside the system unit and press out on the floppy port cover until you can remove it from the front panel.

To install a floppy port cover

Press the cover into the opening in the front panel of the system unit until it snaps into place.



WSMXXM032

Figure 11-4. Floppy port cover

Removing and Installing an Internal Hard Disk Drive

One hard disk drive can be installed in the system unit enclosure as shown in Figure 11-5. The following drive types can be installed internally:

- RZ23L (RZ23L-FN)
- RZ24 (RZ24-FN)
- RZ25 (RZ25-EP)
- RZ26 (RZ26-EP)

SCSI drives of other types are installed externally. For information about externally mounted SCSI drives, refer to the *TURBOchannel Maintenance Guide*.

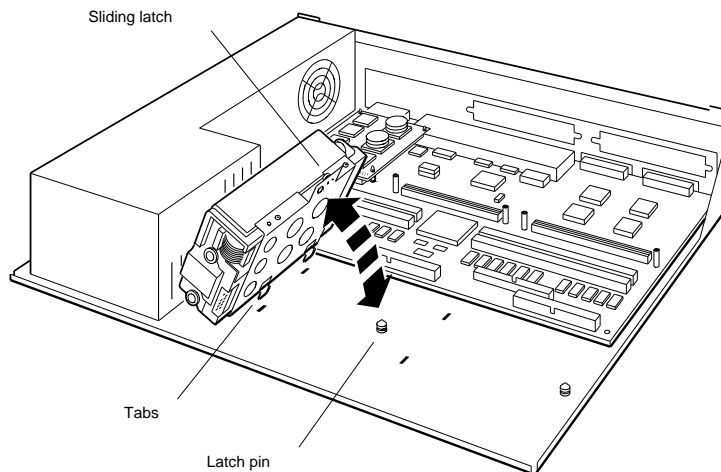
The internal hard disk drive is installed in front of the base system module next to the power supply. The internal SCSI cable (17-01626-05) connects the hard disk drive to the internal SCSI connector on the base system module. The internal drive power cable (17-03357-01) conducts DC power to the hard disk drive from the internal drive power connector on the base system module.

To remove an internal hard disk drive

1. Disconnect the SCSI and power cables from the system module.
2. Release the sliding latch that fastens the drive to the chassis.
3. Tilt the drive assembly toward the power supply, allowing it to pivot around the tabs in the slots. When the tabs are free, lift the drive assembly up and out of the system unit chassis.
4. Disconnect the SCSI and power cables that are attached to the drive.

To install an internal hard disk drive

1. Make sure the drive has the correct SCSI ID. To find the SCSI ID for a specific drive, see the discussion of that drive in “ID Settings for Internal Drives” section of this guide.
2. Connect the internal SCSI cable and internal power cable to the drive.
3. Position the drive assembly over the system unit chassis with the two tabs pointed down into the slots in the chassis. Slide the tabs into the slots.
4. Lower the drive assembly to the floor of the chassis.
5. Engage the sliding latch to fasten the drive assembly in place.
6. Connect the SCSI and power cables to the system module. Be sure to keep the pull tab on the SCSI cable connector away from the SCSI connector on the system module when you connect the SCSI cable.



WSMXM033

Figure 11-5. Removing and installing an internally mounted hard disk drive

ID Settings for Internal Drives

This section describes ID settings for the internal drives the system supports. Labels on top of the drives identify the drives.

The system supports the internally mounted drives listed in Table 11-2:

Table 11-2. Internal Drive ID Settings

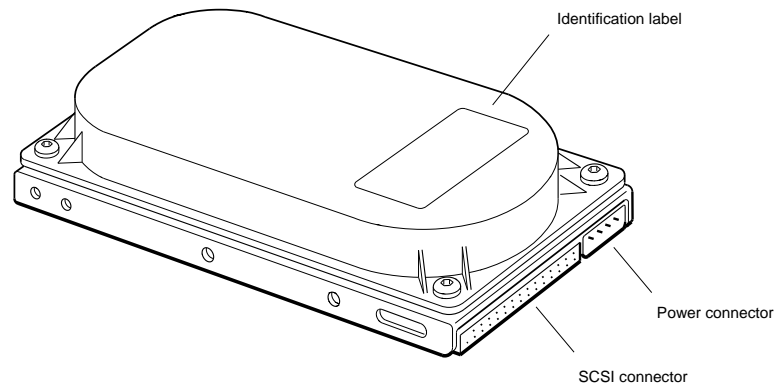
Drive	Default SCSI ID
RZ23L half-height hard disk drive	2
RZ24 hard disk drive	2
RZ25 hard disk drive	2
RZ26 hard disk drive	2
RX26 floppy drive	N/A*

* The RX26 floppy drive is not a SCSI device. However, it has a preset drive select number of 0.

Setting SCSI IDs

To set the SCSI ID of the three hard disk drives available for installation inside the Personal DECstation/DECsystem, use a row of small pins and jumper blocks. Set the ID by moving the jumpers to different positions on the pins.

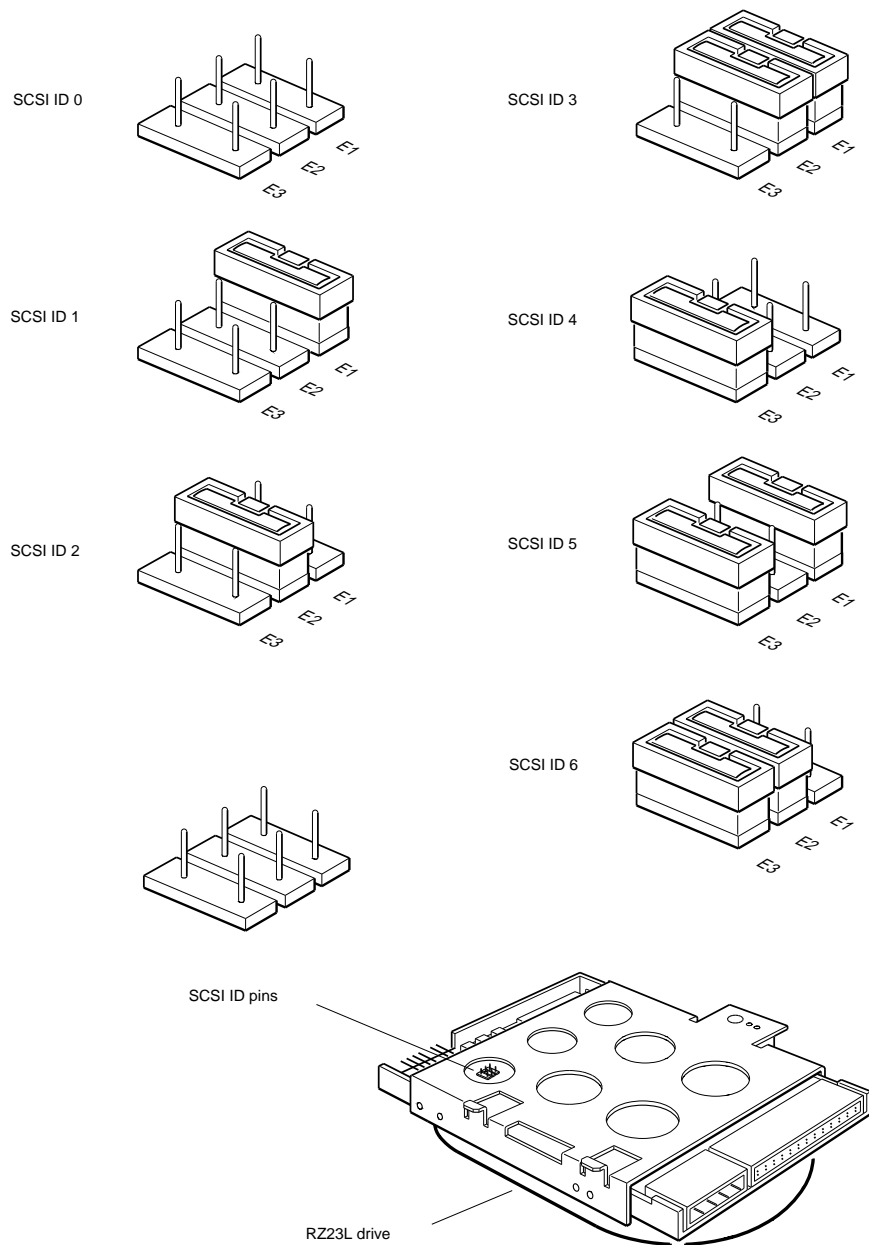
Setting the SCSI ID on an RZ23L hard disk drive



WSMXM035

Figure 11-6. RZ23L half-height hard disk drive

The SCSI jumpers are on the underside of the RZ23L drive. Position the jumpers as shown in Figure 11-7 to set the SCSI ID.



WSMXM034

Figure 11-7. RZ23L SCSI ID jumper settings

Setting the SCSI ID on an RZ24 hard disk drive

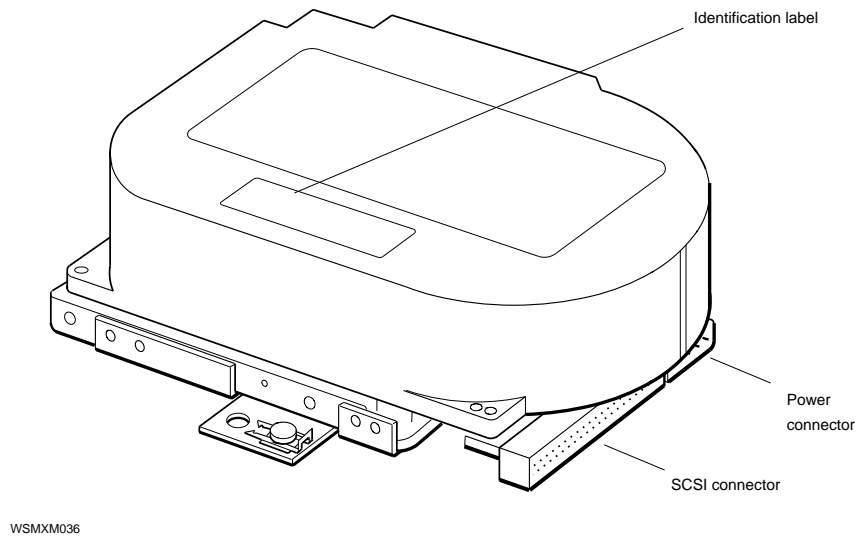


Figure 11-8. RZ24 hard disk drive

The SCSI jumpers are on the underside of the RZ24 drive. Position the blocks as shown in Figure 11-9 to set the SCSI ID.

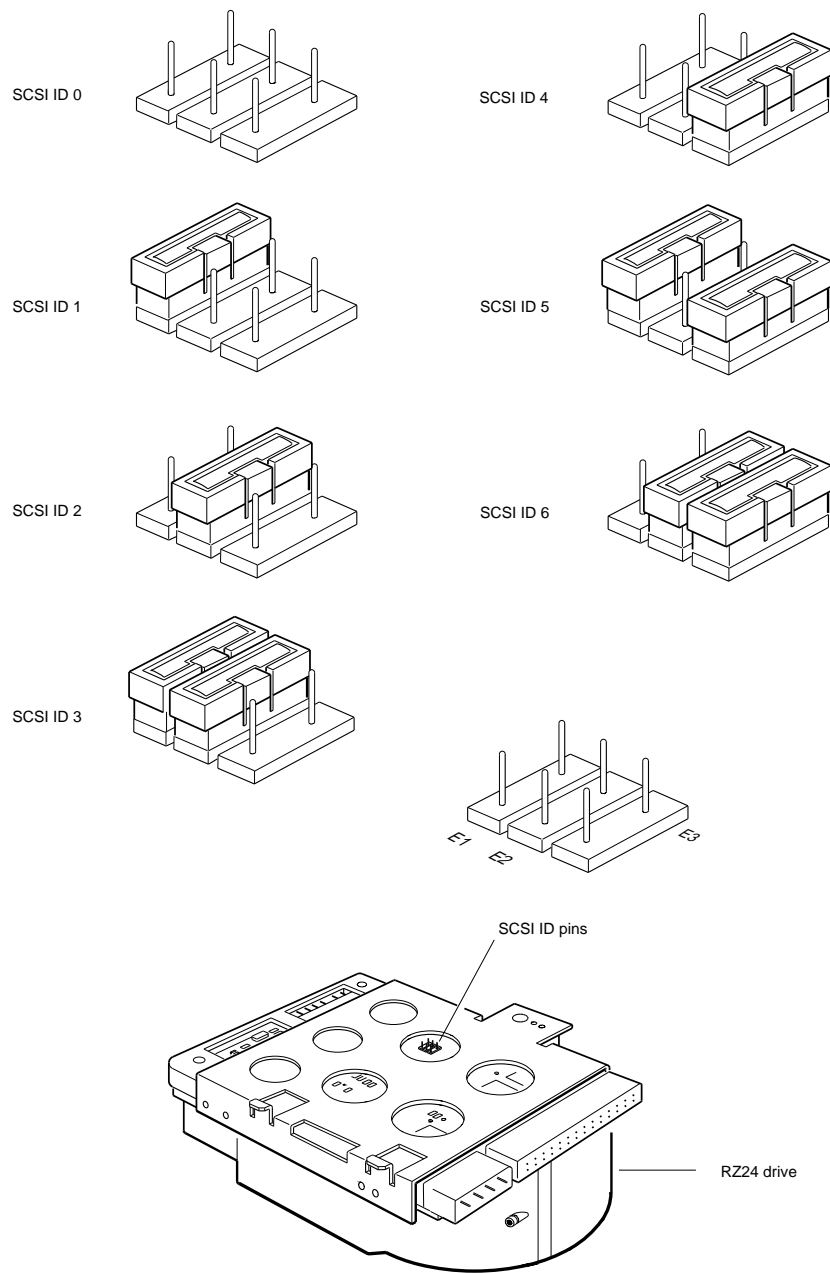


Figure 11-9. RZ24 drive SCSI ID jumper settings

Setting the SCSI ID on an RZ24L hard disk drive

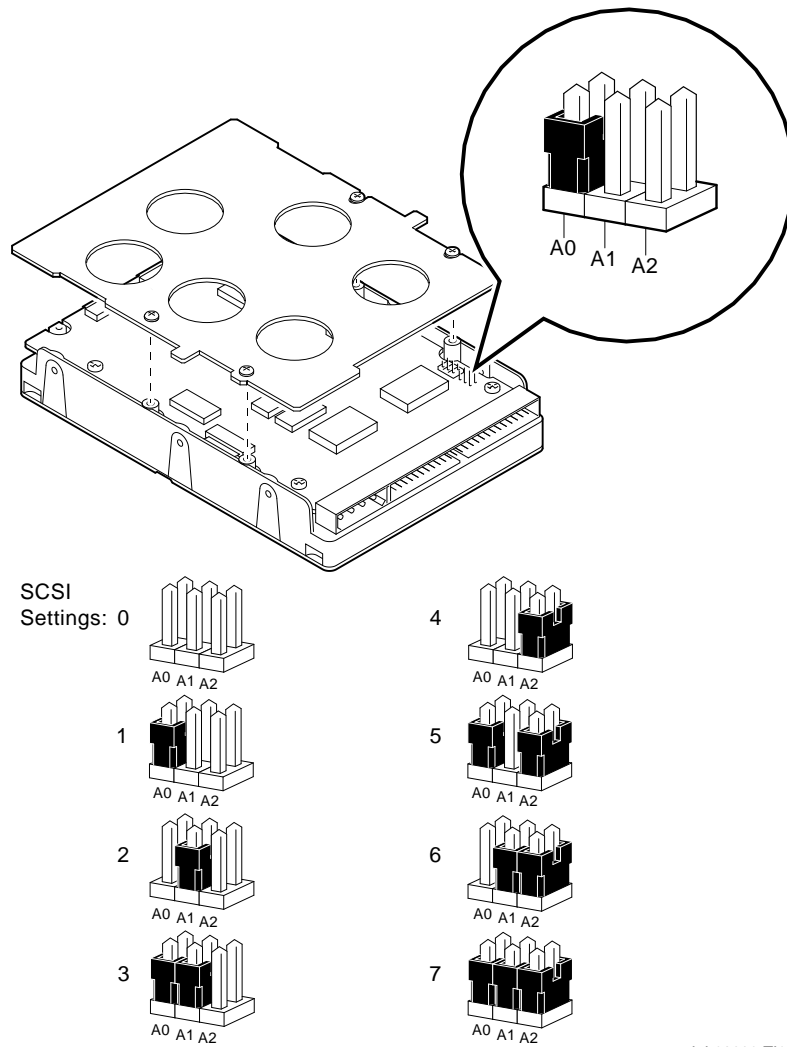


Figure 11-10. RZ24L hard disk drive

LJ-02990-T10

Setting the SCSI ID on an RZ25 hard disk drive

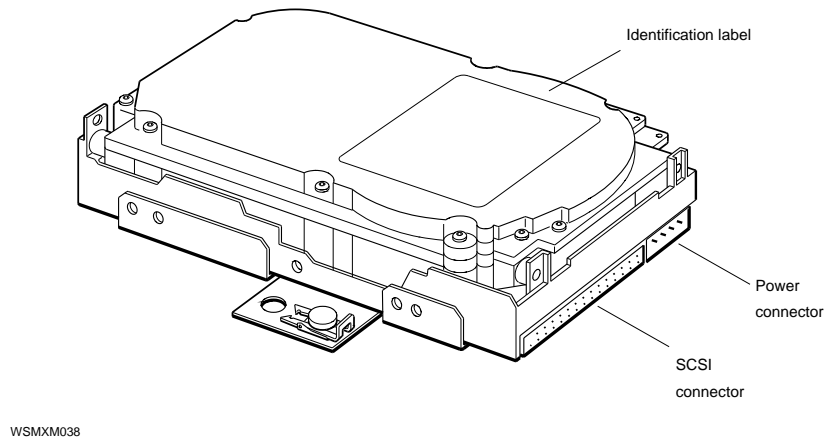


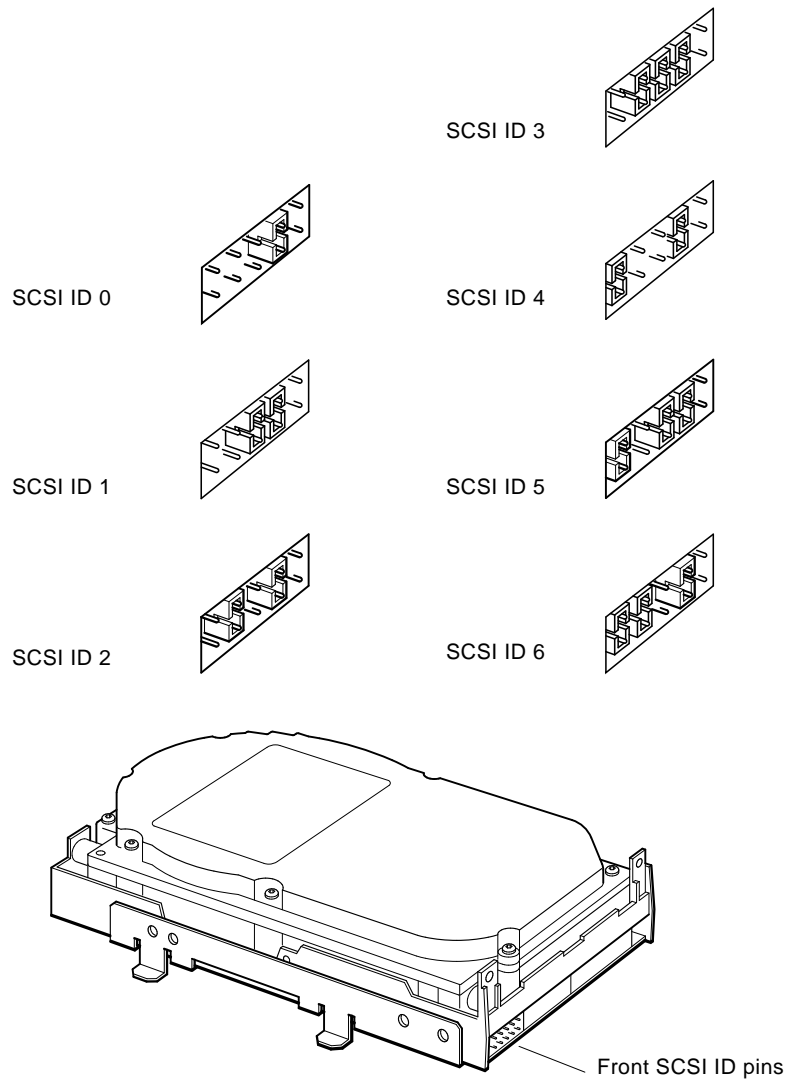
Figure 11-11. RZ25 hard disk drive

The RZ25 drive has SCSI ID pins in two locations, on the front and on the side of the drive. The side pins are hidden during use, and visible only when the drive is removed from its metal enclosure. Use jumpers on either the front pins or the side pins, but not on both.

Figure 11-12 shows the drive with jumpers on the front pins. Figure 11-13 shows the drive removed from its metal enclosure so the jumpers on the side pins are visible. Note that in both illustrations there is a jumper on the fourth pair of pins from the left. This jumper is required whether the front or side pins are used for SCSI ID.

Note: *The front and side settings are different.*

The two figures also show jumper settings for the front and side pins. Note that the front and side settings are different. If you use the front pins, be sure that there are no jumpers on the side pins. If you use the side pins, be sure that the only jumper on the front pins is the one on the fourth pair.



LJ-02960-T10

Figure 11-12. RZ25 drive front SCSI ID jumper settings

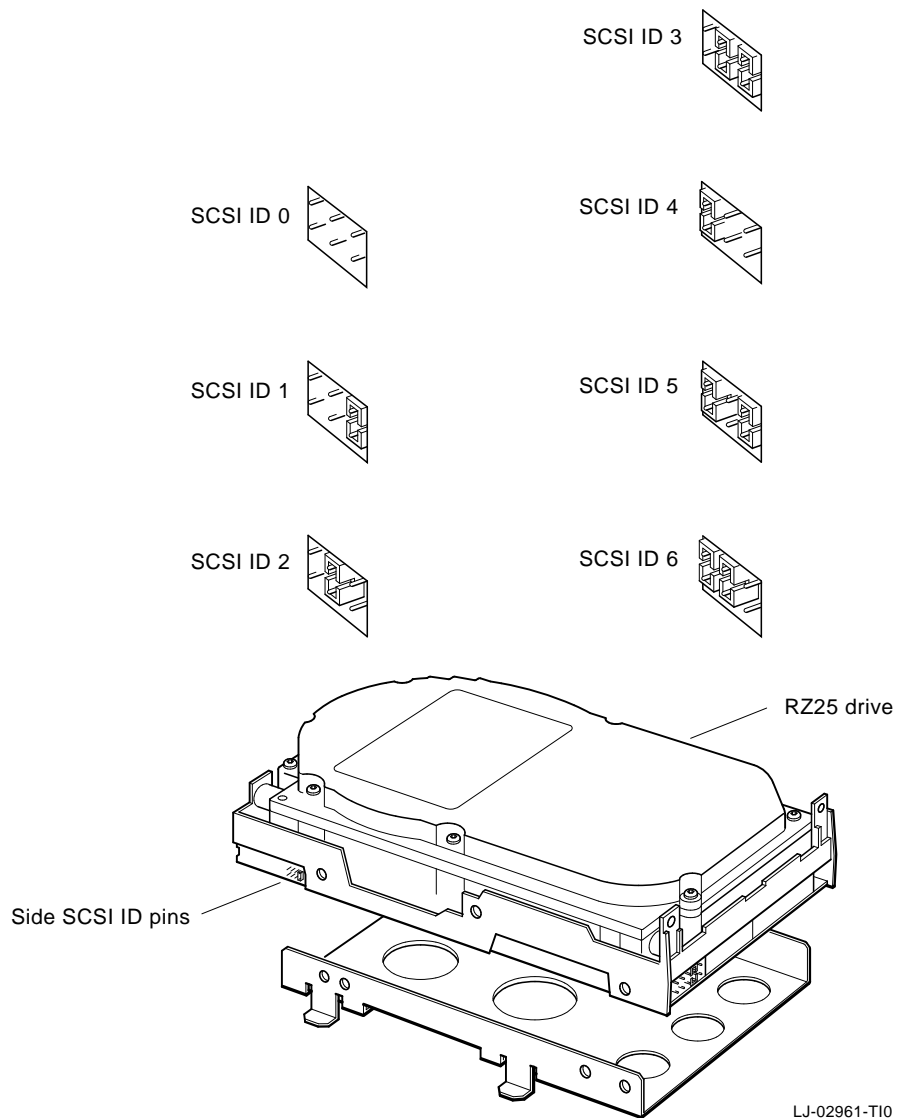
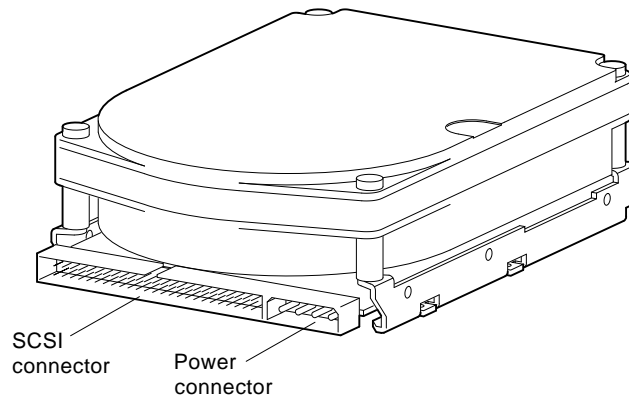
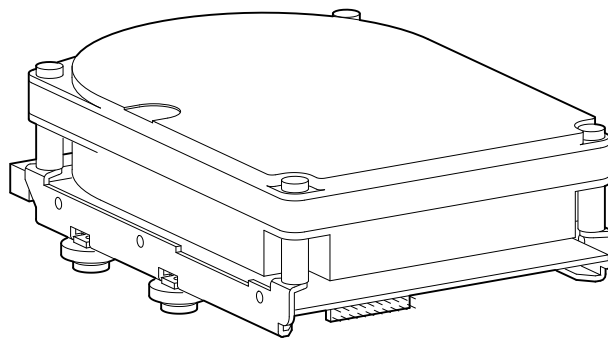


Figure 11-13. RZ25 drive side SCSI ID jumper settings

Setting the SCSI ID on an RZ26 hard disk drive



Front view



Back view

LJ-02980-T10

Figure 11-14. RZ26 hard disk drive front and back view

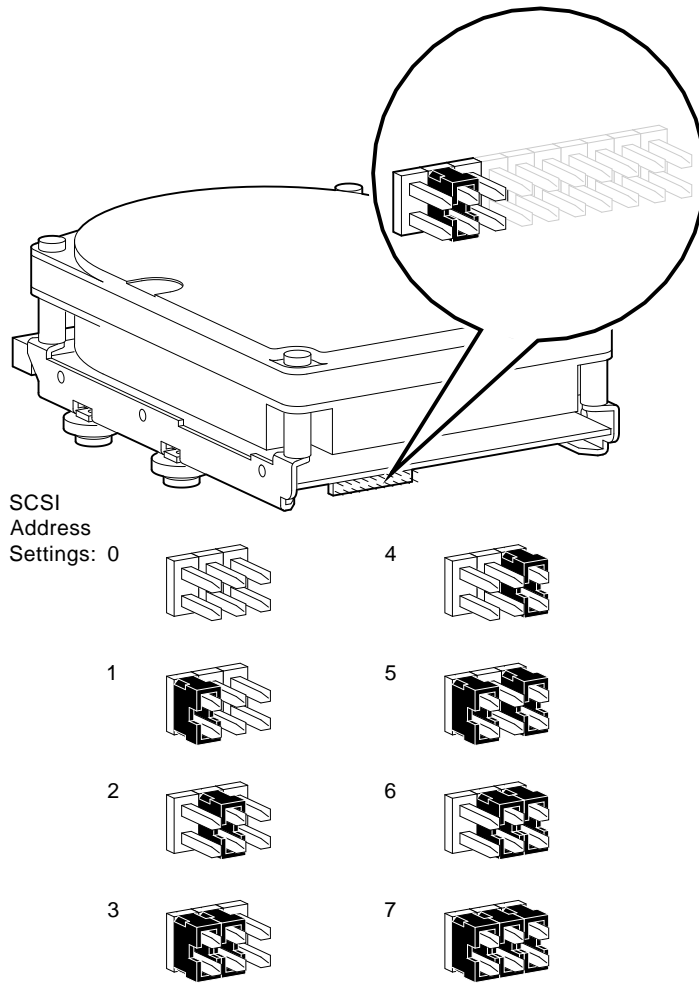


Figure 11-15. RZ26 hard disk drive

Setting the SCSI ID on an TLZ06 tape drive

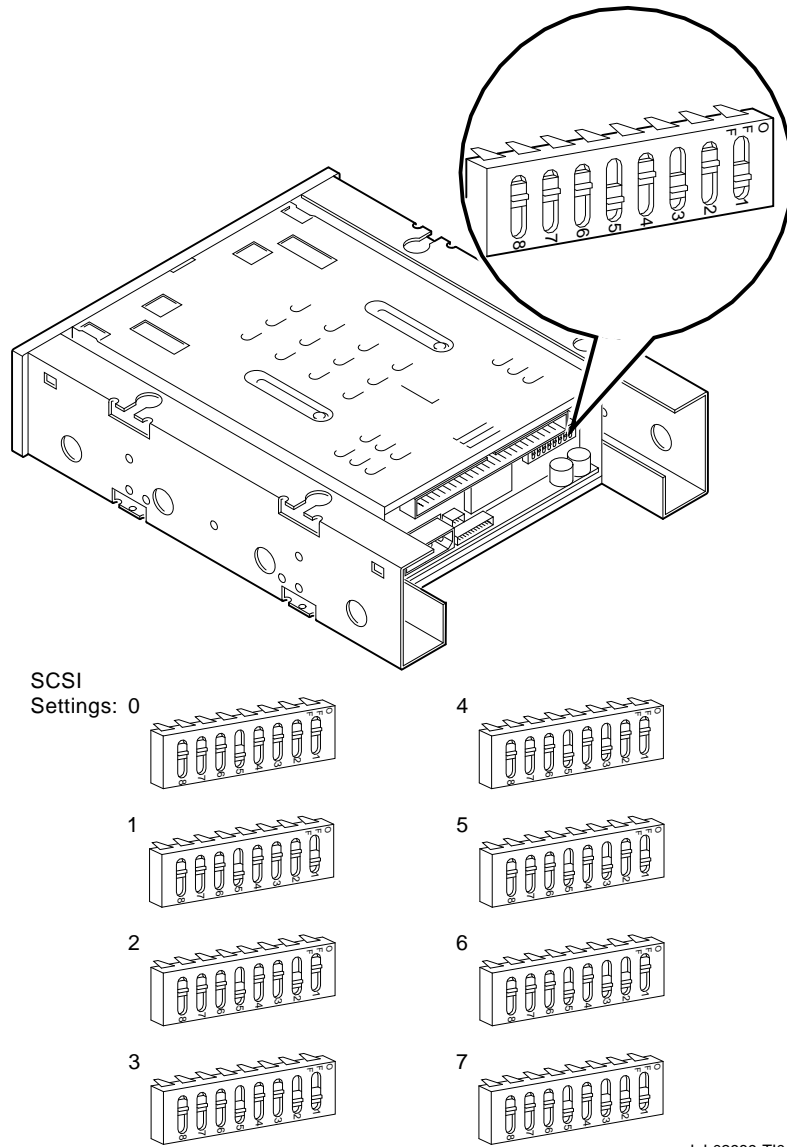
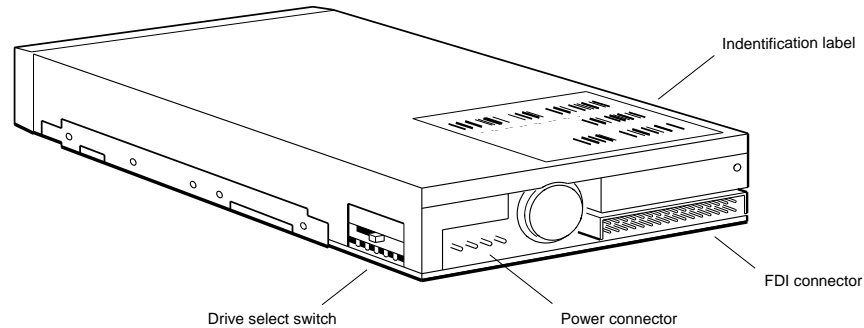


Figure 11-16. TLZ06 cassette tape drive

Setting the drive select switch on an RX26 floppy disk drive

The drive select switch on an RX26 floppy disk drive must be set to 0.



WSMXM070

Figure 11-17. RX26 floppy disk drive

To set the drive select switch,

1. Remove the drive from its mounting and from its perforated floppy drive shield.
2. The drive select switch is toward the rear. Small numbers are printed on the green surface next to the switch. Move the switch to select the number.

For Further Information

For information about individual SCSI drives, refer to the documentation for the specific drives and for the expansion boxes that contain those drives.

For information about TURBOchannel SCSI controller option modules, refer to the *TURBOchannel Option Module Maintenance Guide*.

For information about the SCSI drive terminator, see “Installing and Removing a SCSI Drive Terminator” in Chapter 12 of this guide.

For information about the SCSI controller terminator, see “Installing and Removing a SCSI Controller Terminator” in Chapter 12 of this guide.

For information about troubleshooting the SCSI controllers and drives, see Chapters 4, 5, 6, 7, and 9 of this guide.

CPU, TURBOchannel Options, and Memory

This chapter describes how to remove and install the following field-replaceable units (FRUs):

- The CPU module
- Memory modules
- TURBOchannel option modules
- ESAR chips
- The base system module
- The power supply

Caution: *Always power down equipment before working on equipment.*

Always follow antistatic procedures when handling electronic components. See “Using the Antistatic Kit” in Chapter 10 of this guide.

Removing and Replacing a CPU Module

The CPU module shown in Figure 12-1, uses the MIPS Technologies, Inc R3000A central processing unit (CPU) and R3010 floating point unit (FPU), mounted on the same chip. The CPU uses a 64-kilobyte instruction cache and a 64-kilobyte write-through data cache.

The CPU module shown in Figure 12-2 uses a R4000 Family MIPS Technologies, Inc. central processing unit. This CPU is functionally similar to the R3000, but has support for secondary caching system as well as a 50 MHz clock speed giving the R4000 increased performance over the R3000A. Additionally, the ASIC chip has incorporated a large scale integration design optimizing the interface controlling logic onto a single chip.

The CPU module is replaceable. You can also upgrade to a faster CPU module. The Personal DECstation/DECsystem supports the CPU modules described in Table 12-1.

Table 12-1. CPU Modules

CPU type	FRU number	Customer order number
20-megahertz R3000	54-20615-01	KN02-BC
	54-21337-01*	
25-megahertz R3000	54-20615-02	KN02-CC
	54-21337-02*	
33-megahertz R3000	54-21331-01	KN02-DA
50-megahertz R4000	54-21874-01	KN04-AA

¹These are the newer plastic (as opposed to ceramic) parts; they are functionally equivalent to the earlier variety.

Some CPU modules are similar in appearance. Look on the module for the adhesive label displaying the part number suffix (01 or 02) for help in identifying modules.

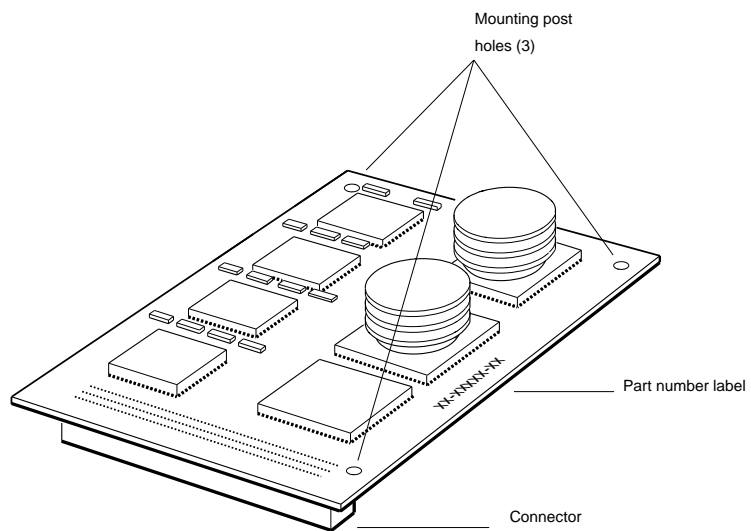


Figure 12-1. R3000 CPU module

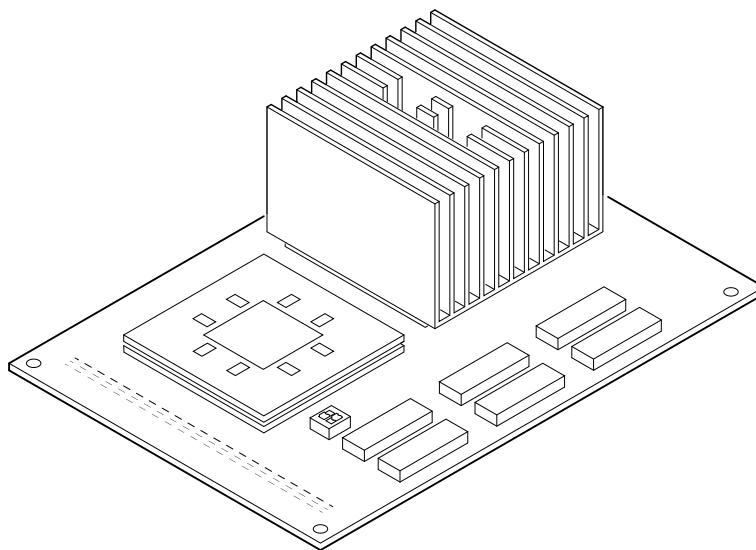


Figure 12-2. R4000 CPU module

To determine which type of module is installed without opening the workstation, run the CPU type test. At the console prompt (>>), type

t 3/misc/cpu-type

and press Return. The CPU type is identified by one of the following messages:

For the 20-megahertz module	3/misc/cpu-type's code: NDX-90A
For the 25-megahertz module	3/misc/cpu-type's code: NDX-96A
For the 33-megahertz module	3/misc/cpu-type's code: NDX-102A
For the 50-megahertz module	3/misc/cpu-type's code: NDX-64A

To Remove a CPU Module

1. Follow all recommended anti-static procedures prior to handling the CPU module.
2. Use your fingers or the PC Removal Tool (74-46254-01) to squeeze each mounting post lock that fastens the CPU module to the base system module as shown in Figure 12-3. Then lift the corner of the CPU module board slightly to free it from the lock.

Directly below the heat sink is a fastener which requires needle-nose pliers to removed the CPU from the fastener.

3. When the board is free of all locks, grip it by the edges at both ends of the connector and lift it up and out of the connector. If necessary, gently rock the module to loosen it.

Caution: *Do not touch the CPU module heat sinks. The heat sinks are fragile and break easily. If the system was recently powered down, they can also be extremely hot.*

To Install a CPU Module

1. Position the CPU module so its connector is aligned over the CPU module connector on the base system module. Align the holes in the CPU module over the mounting posts on the base system module.
2. Press the CPU module connector into the base system module connector.
3. Press the hole in each corner of the CPU module onto the corresponding mounting post lock.
4. Install product conversion label (36-15946-XX) on the top of the base system enclosure, placing the label near the back edge of the cover, centered between the two sides. This label indicates an upgrade was done from a KN02 to a KN04 type CPU card. All upgraded systems should have this label, if it is missing, then put a label on it.

Note: *The product conversion label and the PC Removal Tool are included in the R4000 CPU Module Upgrade Kit.*

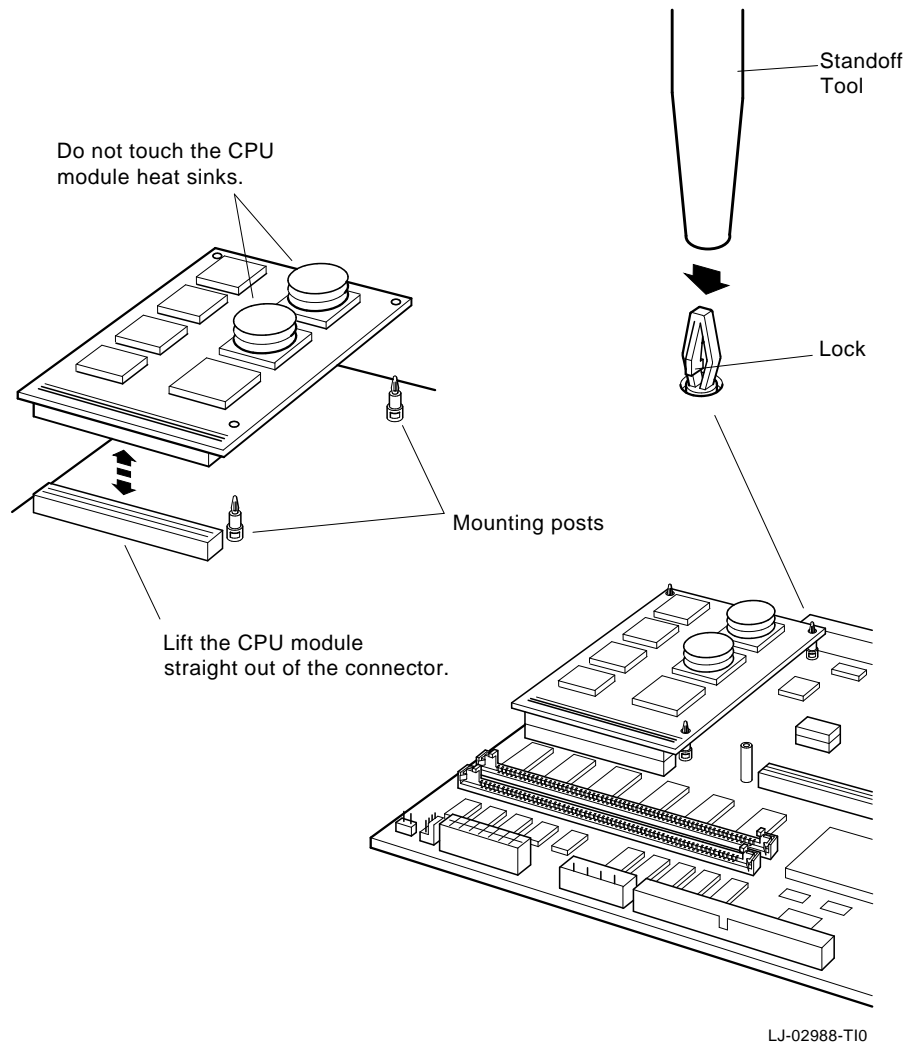


Figure 12-3. Removing and installing a CPU module

Removing and Installing Memory Modules

The base system module memory connectors hold one or two pairs of 2-megabyte or 8-megabyte single in-line memory modules (SIMMs).

- Stand behind the system unit to remove and install memory modules.
 - Reach down in front of the memory modules to release the latches on the front of the connectors with your fingers, not your thumbs.
 - Use the directional references (front, back, left, right) in these instructions.
- Pairs of memory modules must be removed starting with those in the connectors nearest the front of the system unit.
- Memory modules must be installed starting with the pair of connectors nearest the back of the system unit.
- Each pair of connectors is identified by a slot number etched on the edge of the base system module. The connector pair nearest the back of the system unit has slot number 0, and the connector pair nearest the front has slot number 1.
- The memory modules installed in a system must be all 2-megabyte or all 8-megabyte modules. Never combine 2-megabyte and 8-megabyte memory modules in the same system.

Identifying Memory Modules

Table 12-2 shows how to identify the two 2-megabyte modules from each other and the 8-megabyte module.

Table 12-2. Identifying Memory Modules

Customer Order No.	Module Part No.	Size
MS01-AA	57-30735-02	2MB
MS01-CA	57-34320-01	8MB
MS01L-AB	54-21213-01 ¹	2MB

¹The 54-21213-01 (MS01L-AB) memory module can be used interchangeably with the 57-30735-02 (MS01-AA) module and can be used with the DECstation 5000 Model 100 as well as the Personal DECstation but cannot be used with either the 2100 or 3100 models because of base system hardware incompatibility.

Note: *Do not confuse the 50-xxxxx-xx etch number with the module part number which is either a 54-xxxxx-xx or 57-xxxxx-xx class number.*

Figure 12-4 and Figure 12-5 show how to identify the earlier 2MB module versus the 8MB module.

- On the 2-megabyte module, the etch part number (not the module part number) is in the center of the top edge of the back of the module.
- On the 8-megabyte module, the etch part number is on the top left corner on the front of the module.

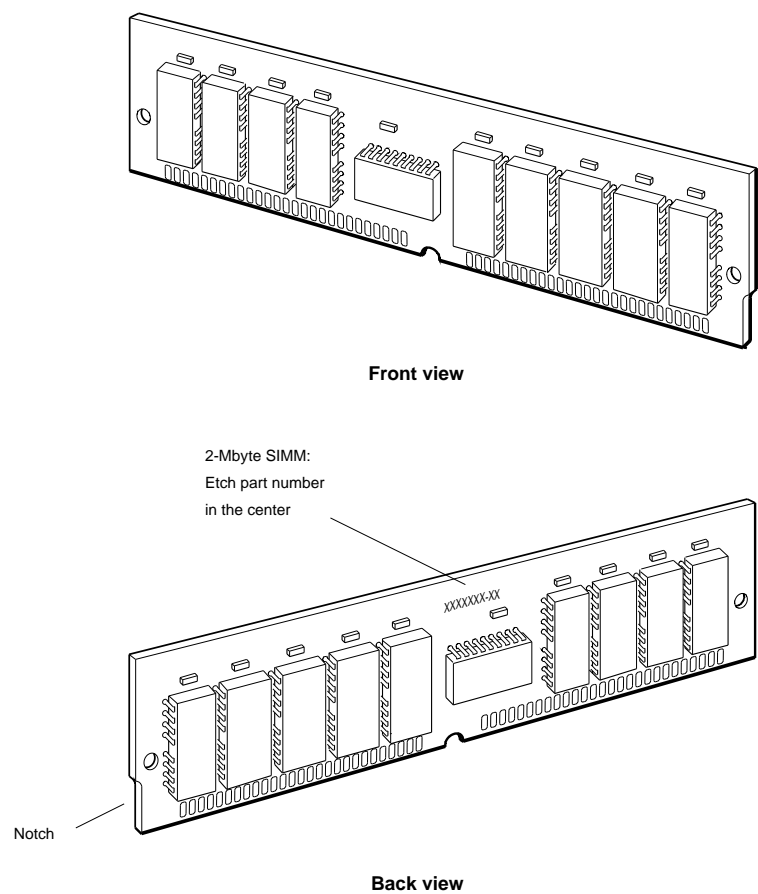
Memory Slot Numbers and Address Ranges

Figure 12-6 shows the memory module slot numbers. These slot numbers are used to identify the memory modules in test commands and error messages.

Table 12-3 shows the address range for each memory module slot and the baseboard RAM. When a memory error occurs, the error message contains the address of the error. You can identify the bad memory module by the address.

Table 12-3. RAM address ranges

Memory Slot	2-Megabyte Module Address Ranges	8-Megabyte Module Address Ranges
0	0x00000000 to 0x003FFFFFFF	0x00000000 to 0x007FFFFFFF
1	0x00400000 to 0x007FFFFFFF	0x00800000 to 0x00FFFFFFF
Baseboard RAM	0x00800000 to 0x00F00000	0x01000000 to 0x01800000



WSMXM021

Figure 12-4. Identifying 2-megabyte memory modules

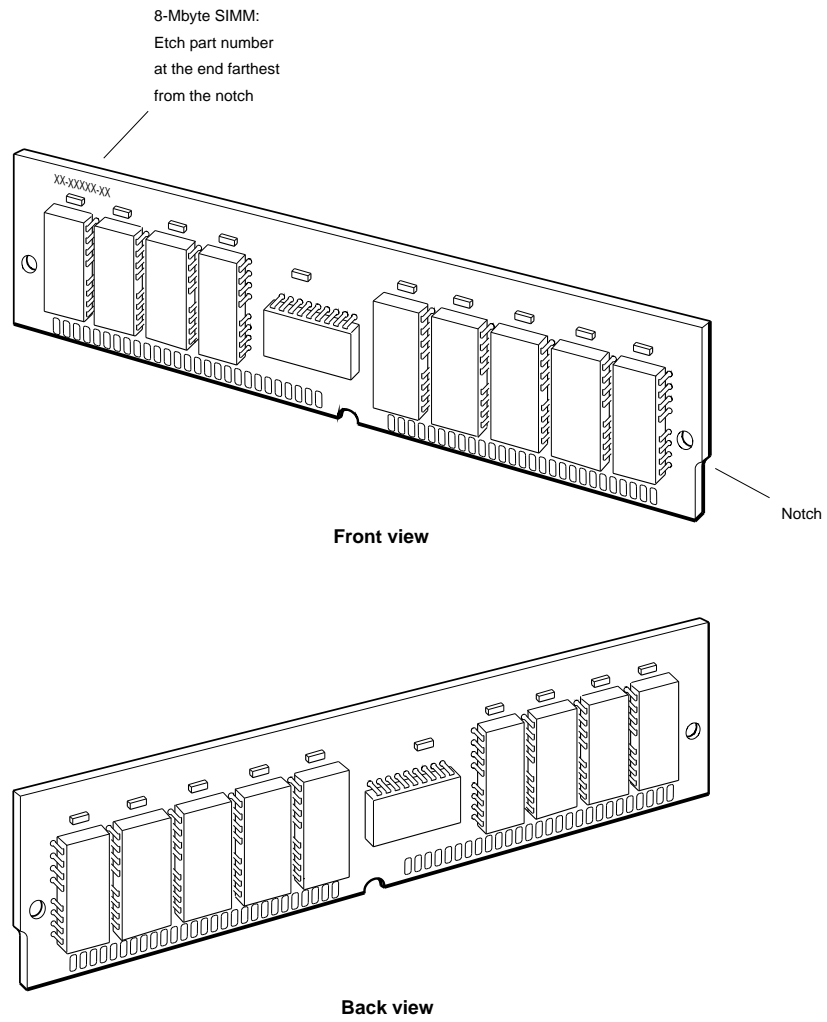
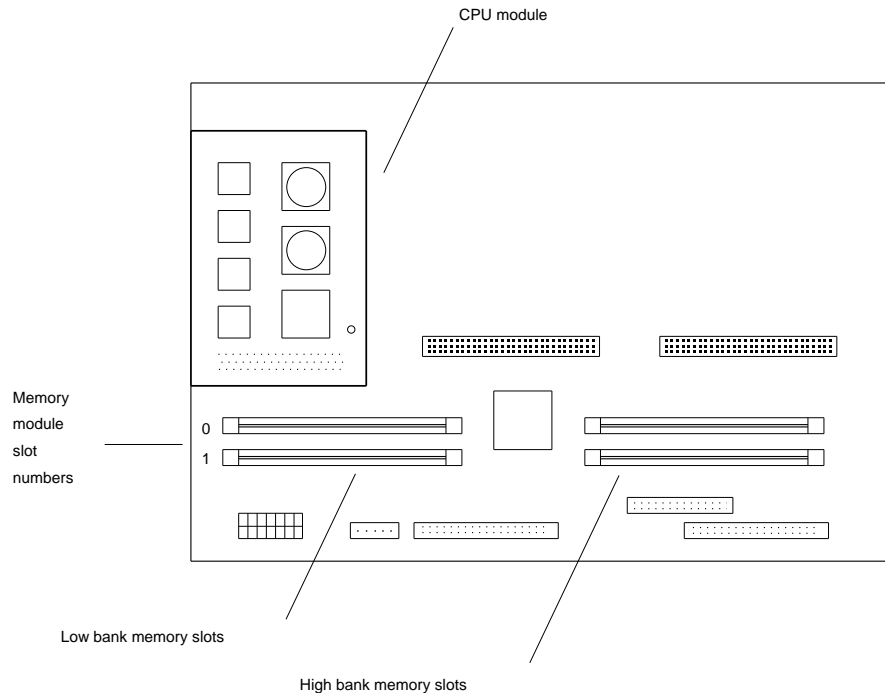


Figure 12-5. Identifying 8-megabyte memory modules



WSMXM023

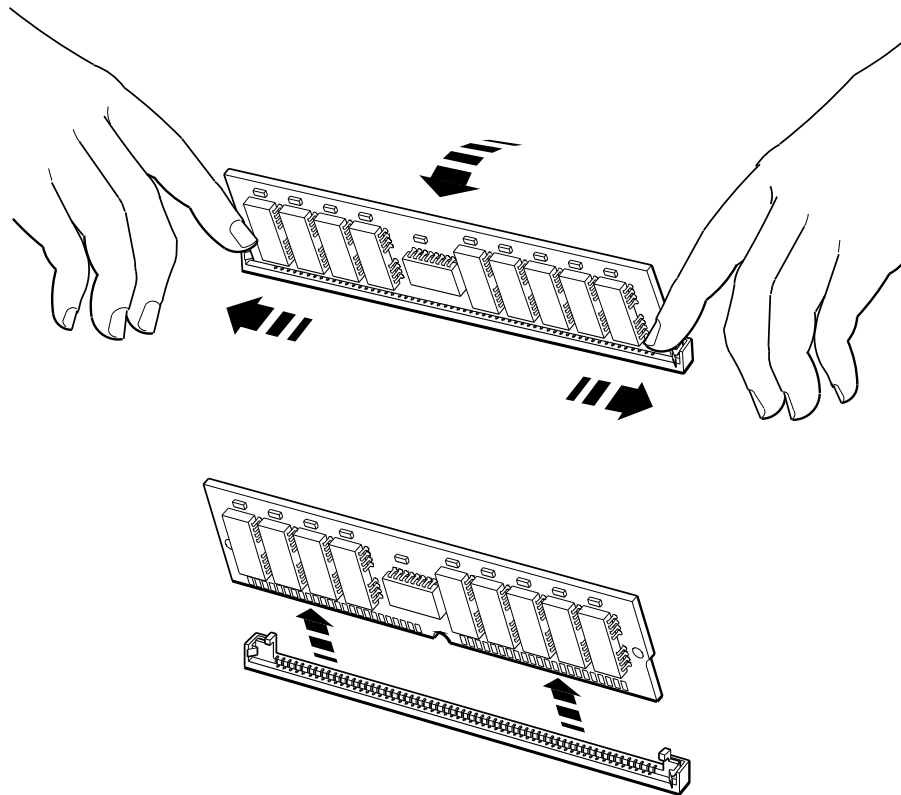
Figure 12-6. Memory module slot numbers

To Remove a Memory Module

1. Use the slot numbers printed beside the memory module connectors to identify the pair of connectors that hold the memory modules you want to replace.
2. If you have bank information from an error message, use the bank information to identify the individual memory module of the pair as follows:
 - The slot nearer the power supply is the low memory bank, identified in error messages by the designator D0-15.
 - The slot farther from the power supply is the high memory bank, identified in error messages by the designator D16-31.

- Reach down in front of the memory module and release the spring clips on the front of the connector with your fingers as shown in Figure 12-7. Tilt the module toward the front of the system unit. Lift the memory module up and out of the connector.
- Repeat this procedure to remove each memory module.

Caution: *There is not enough room to remove a memory module unless it is at the front of the line. Remove any memory modules between the module you are replacing and the front of the system unit first.*



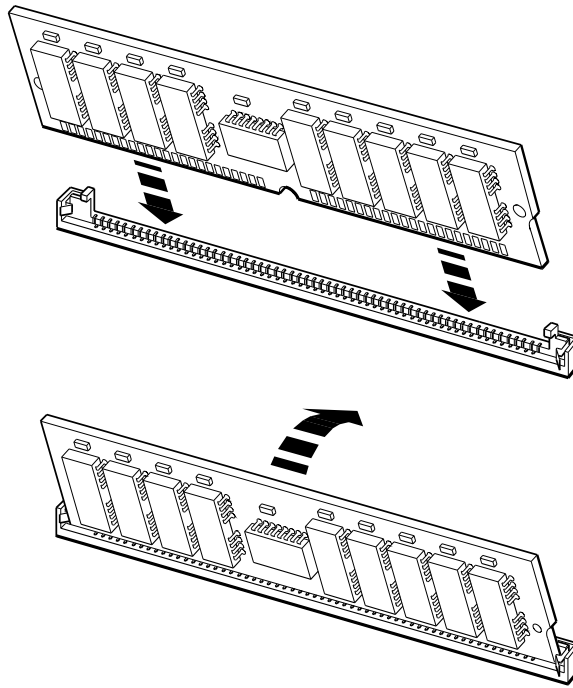
WSMXM024

Figure 12-7. Removing a memory module

To Install a Memory Module

1. Always install memory modules starting with slot 0 or lowest-numbered vacant slot.
2. Position the memory module so the notched end is away from the power supply. Align the lower edge of the memory module over the connector as shown in Figure 12-8.
3. Tilt the top of the memory module toward the front of the system unit and push the connector down into the connector on the base system module.
4. Rotate the memory module to the vertical position, causing the spring clips to snap into place.
5. Repeat the procedure to install each memory module. Remember these modules must be installed in pairs.

Run the **cnfg 3** test to verify that the memory is there. If the power-up self-test reports a home memory error, see Chapter 5 of this guide.



WSMXM025

Figure 12-8. Installing a memory module

Removing and Installing TURBOchannel Option Modules

Caution: Always follow antistatic procedures when handling option modules. Always power down equipment before servicing.

To Remove an Option Module

1. Disconnect any cables that are connected to the rear of the option module.
2. If you are replacing an Ethernet option module, remove the ESAR chip from the old module. See “Ethernet Option Module ESAR Chip” section of this guide and “To Remove and Install a Base System ESAR Chip” section of this guide.

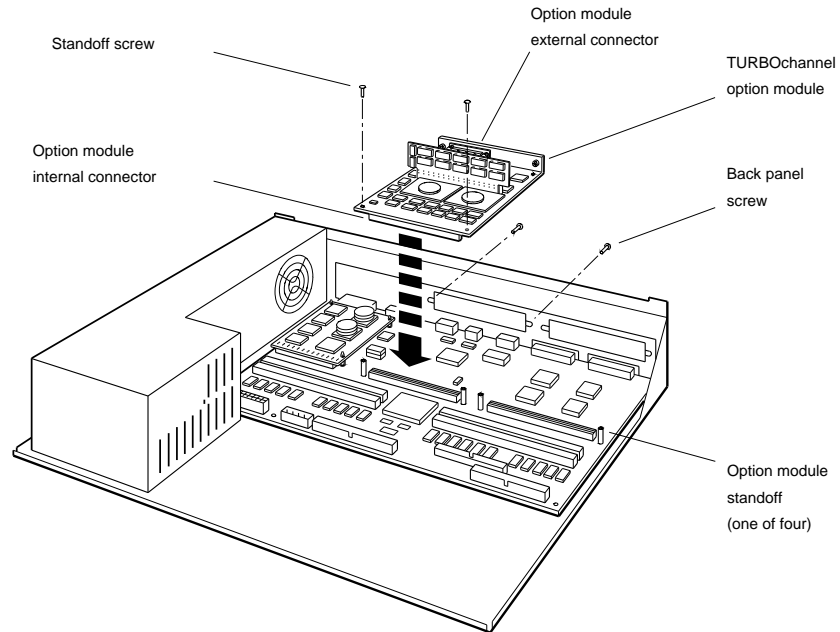
Note: Save the ESAR chip for installation on the new Ethernet option module.

3. Using the system tool, remove the screws that hold the option module to the standoffs and the screws that hold the option module to the back of the system unit as shown in Figure 12-9.
4. Grip the option module board by the edges at both ends of the connector and lift the board up and out of the connector. If necessary, gently rock the module to loosen it.

To Install an Option Module

1. If the rear panel opening for the option module is covered by a blank plate, remove the two screws and then the plate.
2. Position the option module so its external connector is in the appropriate opening on the back of the system unit and its internal connector is aligned over the corresponding TURBOchannel option module connector on the base system module.
3. Firmly press the option module internal connector down into the base system module connector.
4. Insert and tighten the screws that fasten the module to the back of the system unit and the screws that fasten the module to the standoffs.

5. If you are replacing an Ethernet option module, install the ESAR chip from the old module. See “To Remove and Install a Base System ESAR Chip” section of this guide. Insert the chip so the notch on the top of the chip is toward the back of the system unit.



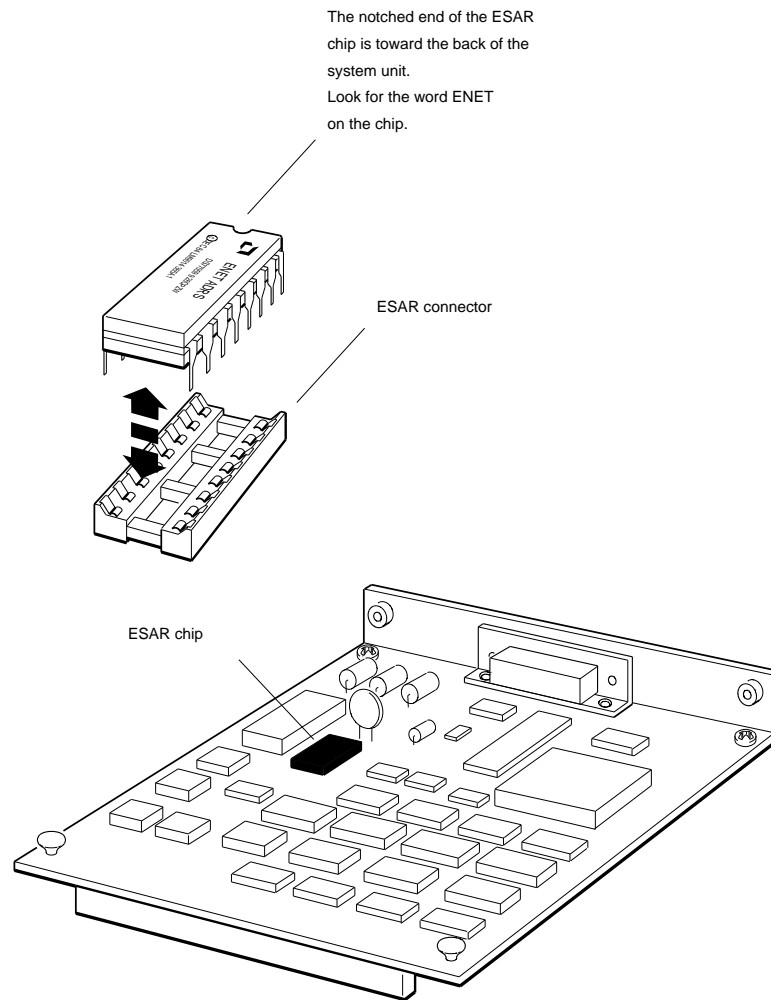
WSMXM027

Figure 12-9. Removing and installing a TURBOchannel option module

Ethernet Option Module ESAR Chip

Every ThickWire Ethernet TURBOchannel option module has an Ethernet Station Address ROM (ESAR) chip (23-365A1-00), which provides the Ethernet address of the Ethernet controller. When you install a new Ethernet option module, remove the ESAR chip from the original base option module and install it on the new one so the workstation will have the same address on the network.

The base system module also has an ESAR chip. See “To Remove and Install a Base System ESAR Chip” section of this guide.



WSMXM028

Figure 12-10. Option module ESAR chip

Base System ESAR Chip

Caution: *Always follow antistatic procedures when handling the electronic components.*

The base system module contains an Ethernet station address register (ESAR) chip (23-365A1-00), which provides the Ethernet address of the base system Ethernet controller. When you install a new base system module, remove the ESAR chip from the old base system module and install it on the new one so that the workstation will have the same address on the network. Every ThickWire Ethernet TURBOchannel option module (PMAD-AB) also has an ESAR chip. (See “Ethernet Option Module ESAR Chip” section of this guide.)

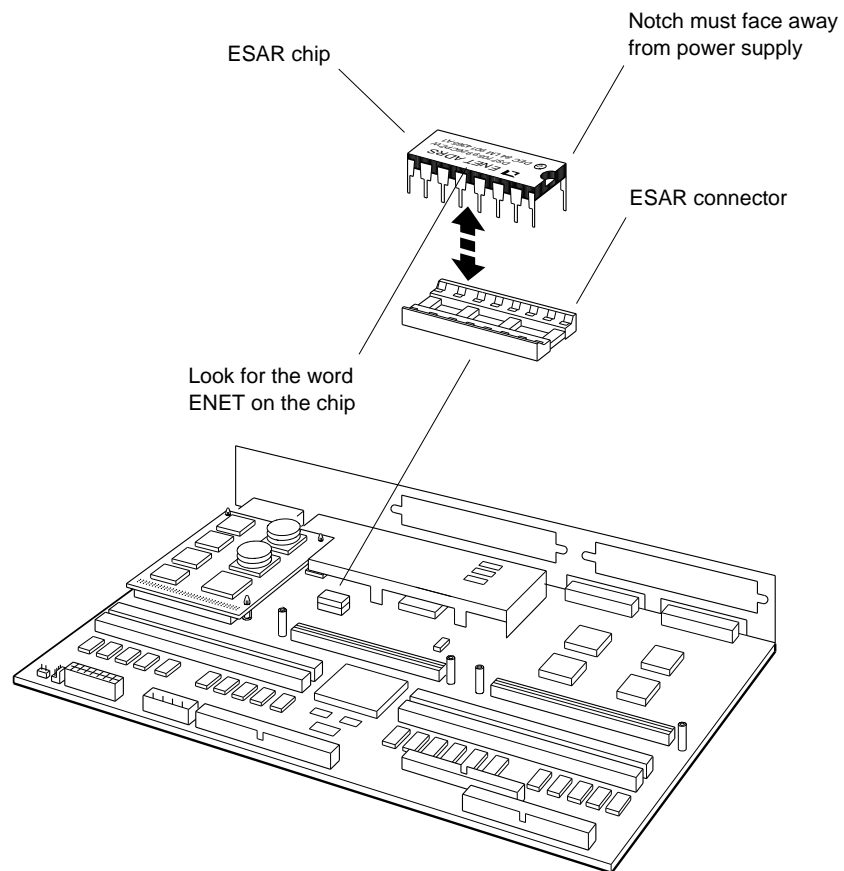
If you use the ESAR chip from the new board, inform the system manager.

To Remove the Base System ESAR Chip

1. If a TURBOchannel option module is installed in slot 1, remove it. See “Removing and Installing TURBOchannel Option Modules” section of this guide.
2. Locate the ESAR chip, shown in Figure 12-11
3. Gently pry the ESAR chip up and out of its socket.

To Install the Base System ESAR Chip

1. Position the chip as shown in Figure 12-11. Make sure the notched end of the chip is away from the power supply and each pin is over a hole in the ESAR chip socket.
2. Press the chip pins into the ESAR socket. Be careful not to bend the pins.



WSMXM018

Figure 12-11. Base system ESAR chip

Removing and Installing the Base System Module

The base system module (54-21325-01) provides basic system functions. The base system module features that are significant to the field maintenance process are listed here and shown in Figure 12-12.

- One CPU module connector
- Eight megabytes of 32-bit parity random access memory (RAM) resident on the system module
- Four connectors for up to 32 megabytes of additional RAM
- One on-board 1024 x 768, 8-plane, 72-hertz color frame buffer video controller
- One SERIAL.bus interface for keyboard and mouse
- One floppy disk interface (FDI) controller for an internally mounted 3.5-inch floppy disk drive (such as the RX26)
- One small computer system interface (SCSI) controller for up to seven drives
- One synchronous/asynchronous RS232 serial communications port with full modem control
- One ThickWire Ethernet controller with an Ethernet station address register (ESAR) chip
- Two connectors for TURBOchannel option modules
- One clear-NVR connector
- Diagnostic ROM

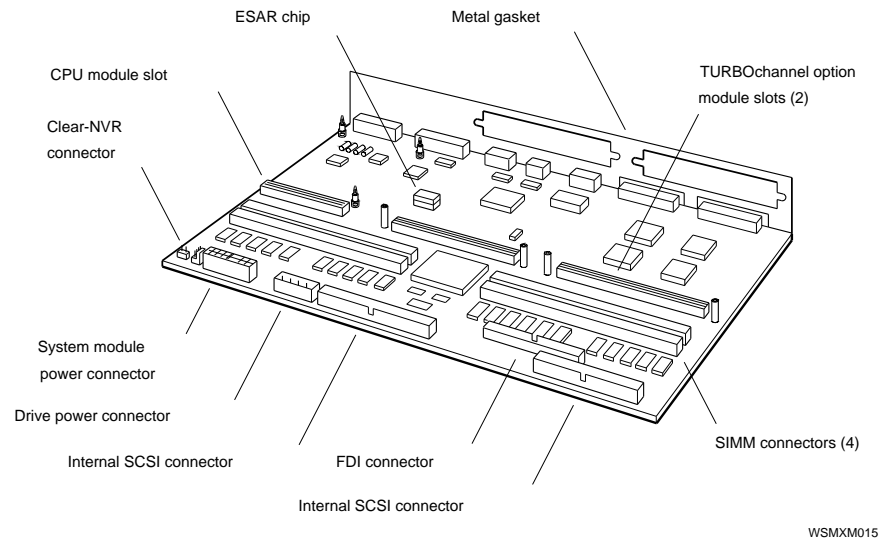


Figure 12-12. Base system module

To Remove the Base System Module

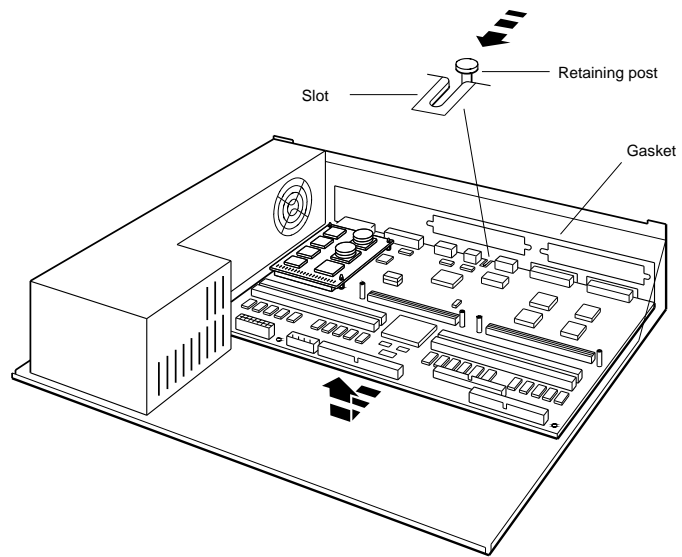
To remove the base system module, you must first remove the following from the system unit:

- The system unit cover (See Chapter 10.)
- The CPU module (See Chapter 11.)
- All memory modules (See Chapter 11.)
- Any TURBOchannel option modules (See Chapter 11.)
- The internal hard disk drive, if there is one (See Chapter 11.)
- The ESAR chip on the base system module (See the description earlier in this chapter.)

Then remove the system module.

1. Disconnect the system module power cord from the system module.
2. Unscrew the following fasteners to release the base system module from the system unit chassis:
 - Seven screws
 - Four CPU module standoffs
 - Four TURBOchannel option module standoffs
3. Slide the system module toward the front of the system unit just enough to clear the retaining post (see Figure 12-13.)
4. Lift the base system module up and away from the chassis.

Caution: *Be careful not to damage the base system module gasket.*



WSMXM016

Figure 12-13. Removing and installing the base system module

To Install the Base System Module

1. Position the system module over the chassis so the metal gasket is up and toward the rear of the chassis.
2. Hold the module against the rear panel of the chassis, compressing the metal gasket. Align the holes in the module over the mounting posts on the chassis. Reinstall the fasteners removed earlier:
 - Seven screws
 - Four CPU module standoffs
 - Four TURBOchannel option module standoffs

Then reinstall the following:

- The ESAR chip
- The internal floppy disk drive, if there is one
- The internal hard disk drive, if there is one
- Any TURBOchannel option modules
- All memory modules
- The CPU module
- The system unit cover
- Connect the system module power cord

Installing and Removing the Power Supply

The power supply (H7817-AA) provides 113 watts of DC power to the system unit. The back panel of the power supply contains the on/off switch, the main power connector, and the convenience outlet. The power supply fan draws air through the system unit enclosure, cooling all components.

The green LED on the front of the power supply is visible through the translucent Digital logo panel on the front panel of the system unit cover. This status indicator is not a standard DCOK LED; it is used solely as a status indicator. This LED turns on only when the power supply is operating correctly.

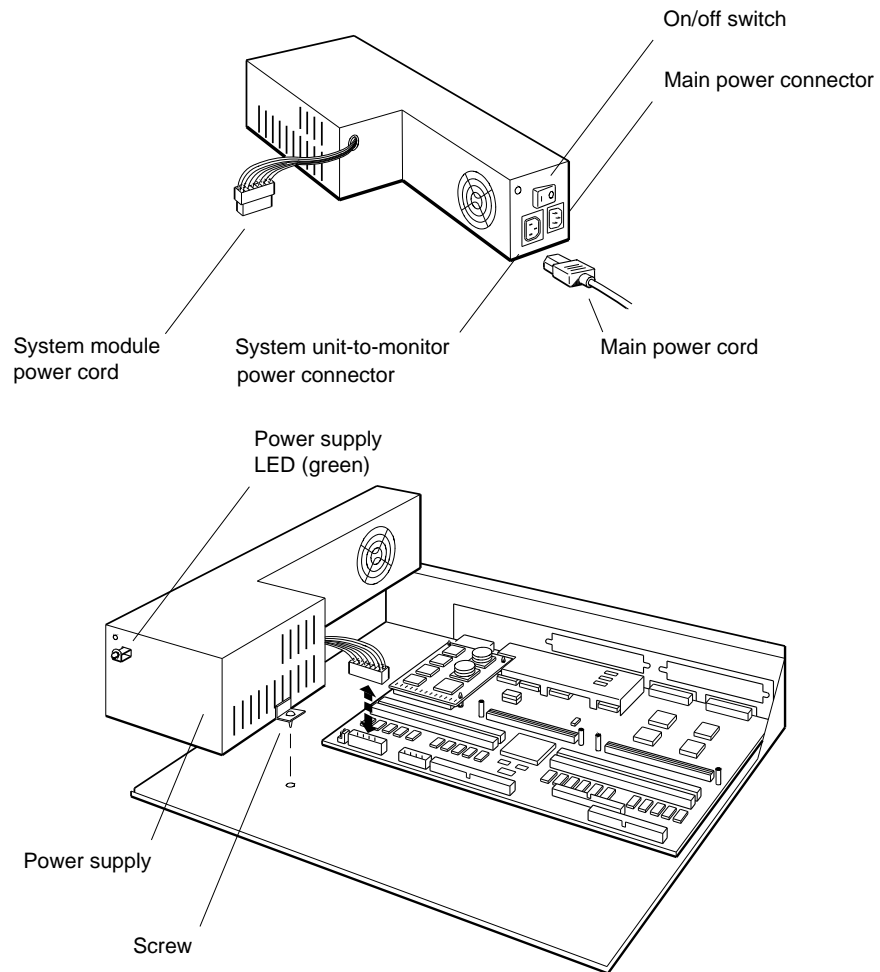
Service of the power supply in the field consists of removing and replacing the entire unit.

To Remove the Power Supply

1. Disconnect the main and accessory power cables from the AC power connectors. Disconnect the system module power cord from the connector on the base system module as shown in Figure 12-14.
2. Remove the screw that fastens the power supply to the floor of the system unit chassis and the screw that fastens the power supply to the rear panel of the system unit chassis.
3. Slide the power supply toward the front of the system unit just enough to clear the rear panel. Raise the rear of the power supply, allowing it to rotate toward the vertical until the tab at the front of the power supply is free in the slot in the floor of the system unit chassis.
4. Lift the power supply up and out of the system unit chassis.

To Install the Power Supply

1. Make sure the system unit power switch is off. (Press the 0 on the switch.)
2. With the rear of the power supply raised, slide the tab on the front of the power supply down into the slot in the floor of the system unit chassis. Lower the rear of the power supply and slide it backward so the rear panel of the power supply is positioned in the opening in the rear panel of the system unit chassis.
3. Install the screw that fastens the power supply to the floor of the system unit chassis and the screw that fastens the power supply to the rear panel of the system unit chassis.
4. Connect the system module power cord to the connector on the base system module.
5. Install the system unit cover.



WSMXM026

Figure 12-14. Power supply

For Further Information

For information about identifying failed FRUs, see Chapters 4, 5, 6, and 9.

For complete information about console commands, see Appendix D.

For information about troubleshooting external drives, SCSI controllers, Ethernet controllers, and TURBOchannel option modules, refer to the *TURBOchannel Maintenance Guide*.

Keyboard and Pointing Devices

The Personal DECstation/DECsystem supports the following input devices.

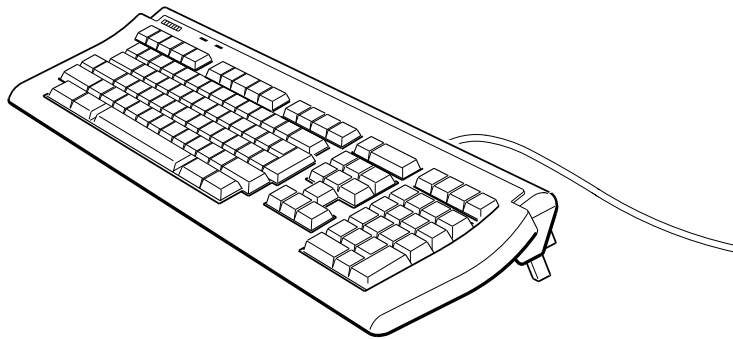
- One keyboard
- One mouse

When the system is configured as a server, the user input and output devices are a terminal with a keyboard connected to the serial communications port.

Keyboard

The system supports the LK501 low-profile SERIAL.bus keyboard.

The keyboard is connected to the SERIAL.bus connector on the rear panel of the system unit by means of the keyboard cable. Connect the mouse cable to one of the connectors recessed in the rear edge of the keyboard and the keyboard cable to the other connector. The two connectors are identical; either cable can be connected to either connector.



LK501 keyboard

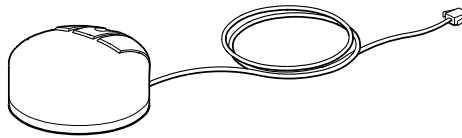
WSMXM040

Figure 13-1. Keyboards

Mouse

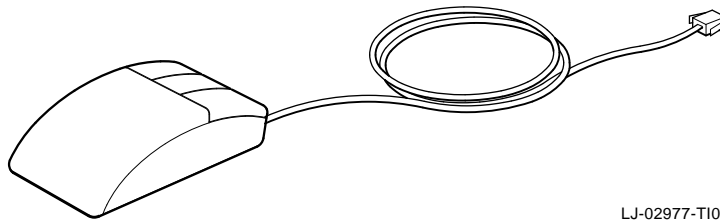
The system supports the VSXXX-BB Figure 13-2 and the VSXXX-GB mouse Figure 13-3.

The mouse connects to the keyboard and the keyboard connects to the SERIAL.bus connector on the rear panel of the system unit by means of the keyboard cable.



WSMXM041

Figure 13-2. VSXXX-BB Mouse



LJ-02977-T10

Figure 13-3. VSXXX-GB Mouse

Removing and Installing the Input Devices

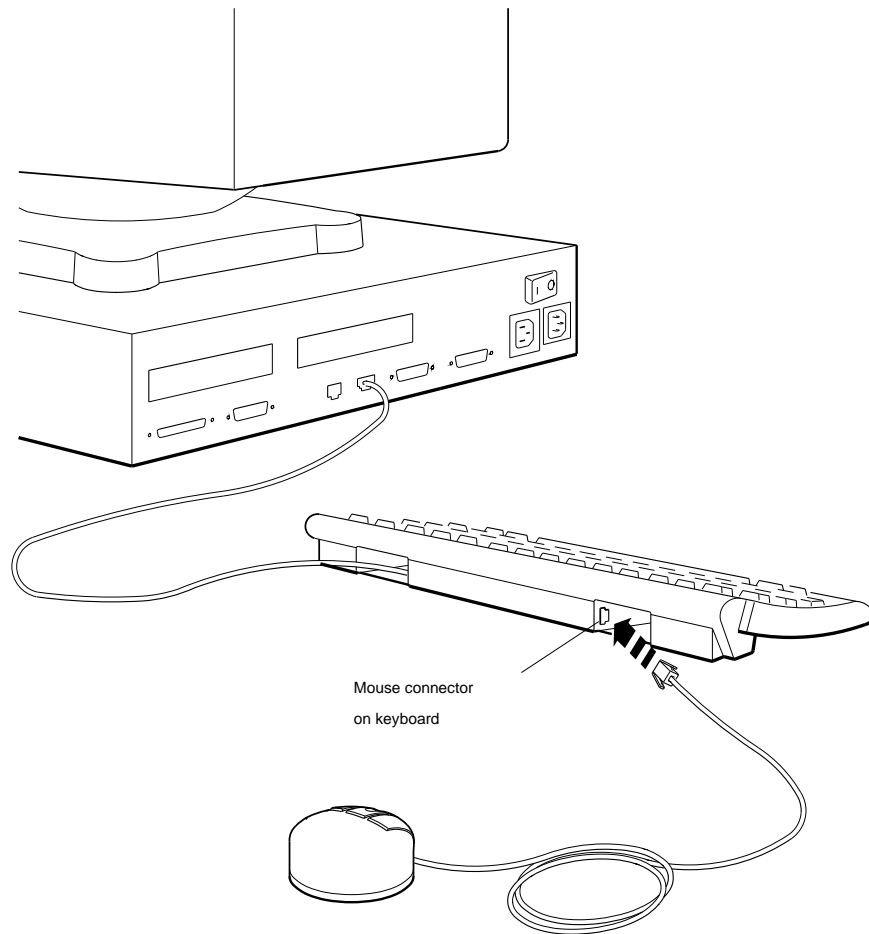
You can remove and replace the keyboard and mouse without resetting the system; that is, you can “hotswap” it.

To Disconnect the Keyboard and Mouse

1. Pinch the two tabs that hold the mouse cable connector on the keyboard connector. Pull the mouse cable connector out of the keyboard connector.
2. Pinch the two tabs that hold the keyboard cable connector in its connector on the keyboard. Pull the keyboard cable connector out of the keyboard connector.
3. Pinch the two tabs that hold the keyboard cable connector in the SERIAL.bus connector on the rear panel of the system unit. Pull the keyboard cable connector out of the SERIAL.bus connector.

To Install the Keyboard and Mouse

1. Push the keyboard cable connector straight into the SERIAL.bus connector on the rear panel of the system unit until the cable connector clicks into place. It fits only one way.
2. Push the cable connector straight into either of the connectors recessed in the rear edge of the keyboard until the cable connector clicks into place. It fits only one way.
3. Push the mouse cable connector straight into either of the connectors recessed in the rear edge of the keyboard until the cable connector clicks into place. It fits only one way.



WSMXM042

Figure 13-4. Removing and installing a keyboard and mouse

Part IV

Appendices

A

Equipment Specifications

This appendix lists the physical specifications, operating conditions, and nonoperating conditions for the following equipment:

- Personal DECstation 5000
- VRC16 monitor
- VRM17 monitor
- LK501 keyboard
- VSXXX-BB mouse
- RZ23L hard disk drive
- RZ24L hard disk drive
- RZ25 hard disk drive
- RZ26 hard disk drive
- TLZ06 cassette tape
- RX26 floppy diskette drive

Personal DECstation 5000 System Unit Equipment Specifications

Table A-1. System Unit Description

Weight	6.80 to 9.50 kg (15.00 to 21.00 lb)
Height	9.14 cm (3.60 in)
Width	40.64 cm (16.00 in)
Depth	43.18 cm (17.00 in)
Input voltage	Auto adjust 100–120 Vac or 220–240 Vac
Input current	1.6 A at 100–120 Vac 0.8 A at 220–240 Vac
Power	
– Frequency	50 to 60 Hz
– Heat dissipation	123 watts, maximum

Table A-2. System Unit Operating Conditions

Temperature range ¹	10°C to 35°C (50°F to 95°F)
Temperature change rate	11°C (52°F) per hour, maximum
Relative humidity	10% to 90%, noncondensing
Maximum wet-bulb temperature	28°C (82°F)
Minimum dew-point temperature	2°C (36°F)
Altitude	2,400 m (8,000 ft) maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-3. System Unit Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Relative humidity	5% to 95%, noncondensing
Maximum wet-bulb temperature	46°C (115°F), packaged
Altitude	4,900 m (16,000 ft) maximum

Table A-4. Acoustics—Preliminary Declared Values per ISO 9296 and ISO 7779

	L_{WA_d}	L_{pAm} (operator position)
Idle	4.5 bels	36 dBA
Operating	4.5 bels	36 dBA

(Current values for specific configurations are available from Digital representatives.)

VRC16 Monitor Equipment Specifications

Table A-5. VRC16 Monitor Description

Weight	25 kg (55.00 lb) maximum
Height	43.18 cm (17 in)
Width	41.15 cm (16.2 in)
Depth	43.43 cm (17.1 in)
External controls, switches, and indicators	Brightness Contrast Power switch Power indicator User interface
Swivel range	$\pm 90^\circ$
Tilt range	-5° to $+15^\circ$
Cathode-ray tube (CRT)	432 mm (17 in) diagonal INVAR shadow-mask, black matrix screen 0.26 mm maximum triad pitch 90° deflection angle 3 color Silica antiglare treatment, purity shielding, dynamic focus correction
Display characteristics	1024 by 768 pixels and 1280 by 1024 pixels Maximum brightness no less than 30 footlamberts (fl)
Video input	
– Termination	75 ohms $\pm 2\%$
– Amplitude	Red: 0.714 Vpp $\pm 2\%$ Green (with sync): 1.0 Vpp $\pm 10\%$ Blue: 0.714 Vpp $\pm 2\%$ Monitor syncs automatically to 66 hz (mode 1) or 72 hz (mode 2)
Horizontal rate timing	
– Active video	13.653 μ s
– Back porch	1.813 μ s

(continued on next page)

Table A-5 (Cont.). VRC16 Monitor Description

– Blanking interval	4.0533 μ s (416 pixels)
– Frequency	56.476 kHz
– Front porch	0.32 μ s
– Horizontal period	17.707 μ s
– Sync pulse	1.920 μ s
Vertical rate timing	
– Active video	13.599 ms
– Blanking interval	0.673 ms
– Frequency	70.069 Hz
– Front porch	0.053 ms
– Sync pulse	0.106 ms
– Vertical period	14.272 ms
Power	
– ac input voltage	100-120 or 220-240 Vac
– Frequency	47 to 63 Hz
– Power consumption	150 watts maximum

Table A-6. VRC16 Monitor Operating Conditions

Temperature range ¹	10°C to 35°C (50° to 95°F)
Relative humidity	10% or less to 95%
Maximum wet-bulb temperature	32°C (89°F)
Minimum dew-point temperature	2°C (36°F)
Altitude	10,000 ft maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-7. VRC16 Monitor Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Relative humidity	90% to 95%
Maximum wet-bulb temperature	66°C (98°F) packaged
Altitude	12,780 m (50,000 ft) maximum

VRM17 Monitor Equipment Specifications

Table A-8. VRM17 Monitor Description

Weight	81.57 kg (37 lb)
Height	41.91 cm (16.50 in)
Width	40.64 cm (16.0 in)
Depth	37.47 cm (14.75 in)
External controls, switches, and indicators	Brightness Contrast Power switch Power indicator High scan/low scan switch
Cathode-ray tube (CRT)	483 mm (431.8mm (17 in)) diagonal Monochrome high efficiency type phosphor
Display characteristics	1280 pixels by 1024 lines (high scan) or 1024 pixels by 768 lines (low scan) Picture size 295 by 236 mm (high scan) or 300 by 225mm (low scan) Maximum brightness no less than footlamberts (fl)
Video input	
– Termination	75 ohms $\pm 2\%$ BNC
– Amplitude	1.0 Vpp composite video
– Refresh rate	72 Hz Factory or field service configurable
Horizontal rate timing	
– Active video time	9.7853 μ s (high scan) and 13.7691 μ s (low scan)
– Back porch	1.7124 μ s (high scan) and 1.7211 μ s (low scan)
– Blanking interval	3.1802 μ s (high scan) and 3.6574 μ s (low scan)
– Frequency	77.1273 kHz (high scan) and 57.384 kHz (low scan)
– Front porch	0.2446 μ s (high scan) and 0.2151 μ s (low scan)

(continued on next page)

Table A-8 (Cont.). VRM17 Monitor Description

– Horizontal period	12.9579 μ s (high scan) and 17.4265 μ s (low scan)
– Pixels displayed	1280 (high scan) and 1024 (low scan)
– Sync pulse	1.2232 μ s (high scan) and 1.7211 μ s (low scan)
Vertical rate timing	
– Back porch	0.427864 ms (high scan) and 0.3834 ms (low scan)
– Blanking interval	0.505658 ms (high scan) and 0.5054 ms (low scan)
– Front porch	0.038897 ms (high scan) and 0.0174 ms (low scan)
– Lines displayed	1024 (high scan) and 76 (low scan)
– Vertical period	13.7842 μ s (high scan) and 13.8889 μ s (low scan)
– Vertical sync pulse	0.038897 μ s (high scan) and 0.1046 μ s (low scan)
– Vertical frequency	72.5562 Hz (high scan) and 72.033 Hz (low scan)
Power	
– ac input	88 to 264 volts
– Frequency	47 to 63 Hz
– Power consumption	75 watts

Table A-9. VRM17 Monitor Operating Conditions

Temperature range ¹	10°C to 35°C (50°F to 95°F)
Temperature change rate	11° (20°F) per hour maximum
Relative humidity	10% to 90% noncondensing
Maximum wet-bulb temperature	28°C (82°F)
Minimum dew-point temperature	2°C (36°F)
Altitude	2400 m (8000 ft) maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-10. VRM17 Monitor Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Relative humidity	95% to noncondensing
Maximum wet-bulb temperature	46°C (115°F) packaged
Altitude	4900 m (16000 ft) maximum

LK501-AA Keyboard Equipment Specifications

Table A-11. LK501-AA Keyboard Description

Weight	1.16 kg (2.56 lb)
Height	4.76 cm (1.88 in)
Width	47.80 cm (19.00 in)
Depth	19.20 cm (7.56 in)
Number of keys	108
Number of indicators	2 status LEDs
Language variations	15 Software selectable (keycaps required)
Cable	1.8 m (6 ft) uncoiled length 4-pin mmj connector at one end
Baud rate	4800
Electrical interface	EIA RS 423
Power consumption	2.0 watts maximum
Power input	12 V \pm 6% at 350 ma
Volume control	8 levels, plus off
Keystroke timing	20 ms minimum

Table A-12. LK501-AA Keyboard Operating Conditions

Temperature range ¹	10°C to 35°C (50°F to 95°F)
Relative humidity	10% to 90% noncondensing
Maximum wet-bulb temperature	32°C (90°F)
Minimum dew-point temperature	15°C (60°F)
Altitude	2,400 m (8,000 ft) maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-13. LK501-AA Keyboard Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Relative humidity	10% to 95%, noncondensing
Maximum wet-bulb temperature	52°C (126°F), packaged
Altitude	4,900 m (16,000 ft) maximum

VSXXX-BB Mouse Equipment Specifications

Table A-14. VSXXX-BB Mouse Description

Weight	0.82 kg (0.37 lb)
Height	39.9 mm (1.57 in)
Diameter	88.9 mm (3.50 in)
Buttons	3
Cable length	1.5 m (5 ft) shielded, 6 conductors and terminals in a 7-pin micro-DIN-type connector (male)
Accuracy	$\pm 3\%$ 0 to 24.5 cm (0 to 10 in) per second in any direction $\pm 15\%$ 24.5 to 49 cm (10 to 20 in) per second in any direction $\pm 30\%$ 49 to 73.5 cm (20 to 30 in) per second in any direction
Baud rate	4800
Data format	Delta binary
Electrical interfaces	Serial.bus
Operating modes	Incremental or polling
Resolution	79 counts per cm (200 counts per in)
Tracking speed	73.5 cm (30 in) per second
Tracking rate	In incremental mode: 55 reports per second In polling mode: up to 95 reports per second

Table A-15. VSXXX-BB Mouse Operating Conditions

Temperature range ¹	10°C to 35°C (50°F to 95°F)
Relative humidity	10% to 90% noncondensing
Maximum wet-bulb temperature	28°C (82°F)
Minimum dew-point temperature	2°C (36°F)
Altitude	2,400 m (8,000 ft) maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-16. VSXXX-BB Mouse Nonoperating Conditions

Temperature range	40°C to 66°C (40°F to 151°F)
Relative humidity	5% to 95% noncondensing
Maximum wet-bulb temperature	46°C (115°F) packaged
Altitude	4,900 m (16,000 ft) maximum

RZ23L SCSI Hard Disk Drive Equipment Specifications

Table A-17. RZ23L SCSI Hard Disk Drive Description

Internal drive	
Weight	0.54 kg (1.20 lb)
Height	2.54 cm (1.00 in)
Width	10.16 cm (4.00 in)
Depth	14.61 cm (5.75 in)
Capacity	
– Bytes per drive	121.65 MB
– Blocks per drive	237,588
– Block size	512 bytes
Data transfer rate	
– Bus asynchronous mode	3.0 MB per second
– Bus synchronous mode	4.0 MB per second
– To and from media	1.5 MB per second
Seek time	8 ms track-to-track
	19 ms average
	35 ms maximum
Average latency	8.8 ms
Interface	SCSI

Table A-18. RZ23L SCSI Hard Disk Drive Operating Conditions

Temperature range ¹	10°C to 55°C (50°F to 128°F)
Temperature change rate	11°C (20°F per hour, maximum
Relative humidity	8% to 80% noncondensing
Maximum wet-bulb temperature	26°C (78°F)
Minimum dew-point temperature	2°C (36°F)
Altitude	–300 to 4600 m (–1000 to 15,000 ft)

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-19. RZ23L SCSI Hard Disk Drive Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Temperature change rate	20°C (36°F) per hour, maximum
Relative humidity	8% to 95% packaged, noncondensing
Maximum wet-bulb temperature	46°C (115°F) packaged
Altitude	–300 to 12,200 m (–1,000 to 40,000 ft)

RZ24L SCSI Hard Disk Drive Equipment Specifications

Table A-20. RZ24L SCSI Hard Disk Drive Description

Internal drive	
– Weight	0.77 kg (1.70 lb)
– Height	4.14 cm (1.63 in)
– Width	10.16 cm (4.00 in)
– Depth	14.61 cm (5.75 in)
Capacity	
– Bytes per drive	245.4M
– Blocks per drive	479,350
– Block size	512 bytes
Data transfer rate	
– Bus asynchronous mode	3.0 MB per second
– Bus synchronous mode	4.0 MB per second
– To and from media	1.5 MB per second
Seek time	5 ms track-to-track
	16 ms average
	35 ms maximum
Average latency	8.3 ms
Interface	SCSI

Table A-21. RZ24L SCSI Hard Disk Drive Operating Conditions

Temperature range ¹	10°C to 55°C (50°F to 131°F)
Temperature change rate	11°C (20°F) per hour, maximum
Relative humidity	8% to 80% noncondensing
Maximum wet-bulb temperature	26°C (78°F)
Altitude	–300 to 4600 m (–1000 to 15,000 ft)

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-22. RZ24L SCSI Hard Disk Drive Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Relative humidity	8% to 95% packaged, noncondensing
Maximum wet-bulb temperature	46°C (115°F) packaged
Altitude	–300 m to 12,200 m (–1,000 ft to 40,000 ft)

RZ25 SCSI Hard Disk Drive Equipment Specifications

Table A-23. RZ25 SCSI Hard Disk Drive Description

Internal drive	
– Weight	0.82 kg (1.8 lb)
– Height	4.14 cm (1.63 in)
– Width	10.16 cm (4.00 in)
– Depth	14.61 cm (5.75 in)
Capacity	
– Bytes per drive	426 MB
– Blocks per drive	832,527
– Block size	512 bytes
Data transfer rate	
– Bus asynchronous mode	3.0 MB per second
– Bus synchronous mode	4.0 MB per second
– To and from media	2.1–3.2 MB per second
Seek time	2.5 ms track-to-track
	14 ms average
	26 ms maximum
Average latency	6.8 ms
Interface	SCSI

Table A-24. RZ25 SCSI Hard Disk Drive Operating Conditions

Temperature range ¹	10°C to 55°C (50°F to 131°F)
Temperature change rate	11°C (20°F) per hour, maximum
Relative humidity	8% to 80% noncondensing
Maximum wet-bulb temperature	26°C (78°F)
Altitude	–300 to 3050 m (–1000 to 10,000 ft)

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft) increase in altitude.

Table A-25. RZ25 SCSI Hard Disk Drive Nonoperating Conditions

Temperature range	–40°C to 66°C (–40°F to 151°F)
Temperature change rate	20°C (36°F) per hour, maximum
Relative humidity	8% to 95% packaged, noncondensing
Maximum wet-bulb temperature	46°C (115°F) packaged
Altitude	–300 m to 12,200 m (–1,000 ft to 40,000 ft)

RZ26 SCSI Hard Disk Drive Equipment Specifications

Table A-26. RZ26 SCSI Hard Disk Drive Description

Internal drive	
– Weight	0.9 kg (1.9 lb)
– Height	4.13 cm (1.625 in)
– Width	10.2 cm (4.00 in)
– Depth	14.6 cm (5.75 in)
Capacity	
– Bytes per drive	1050 M
– Bytes per surface	75M
– Bytes per track	29,640
– Buffer size	512 KB
Data transfer rate	
– To and from media	2.6 MB per second
Seek time	
– Track-to-track	1ms
– Average	10 ms
Average latency	5.6 ms
Rotational speed	5363 rpm

Table A-27. RZ26 SCSI Hard Disk Drive Operating Conditions

Ambient temperature	10°C to 50°C (50°F to 122°F)
Relative humidity	10% to 90% noncondensing

Table A-28. RZ26 SCSI Hard Disk Drive Nonoperating Conditions

Ambient temperature	–40°C to 66°C (–40°F to 151°F)
Relative humidity	8% to 95% packaged, noncondensing

TLZ06 Cassette Tape Drive Equipment Specifications

Table A-29. TLZ06 Cassette Tape Drive Description

Dimensions	
– Weight	2.2 kg (4.7 lb)
– Height	4.1 cm (1.6 in)
– Width	14.6 cm (5.75 in)
– Depth	30.00 cm (12.00 in)
Capacity	1.2
– Per drive	320 MB, approximate DC6320 cartridge
– Per drive	525 MB, approximate DC6525 cartridge
– Per surface	2.48 MB
– Blocks per drive	649,040

Table A-30. TLZ06 Cassettes

Cassette	Size	Capacity
TLZ06-CA	4mm x 90m	2 or 4 GB

Table A-31. TLZ06 Cassette Tape Drive Operating Conditions

Operating temperature	10°C to 40°C (50°F to 104°F)
Operating humidity	20% to 80% noncondensing
Altitude	0 m to 4572 m (0 to 15,000 ft)

Table A-32. TLZ06 Cassette Tape Drive Nonoperating Conditions

Nonoperating temperature	–40°C to 70°C (40°F to 158°F)
Relative humidity	5% to 95% noncondensing
Altitude	0 m to 15,240 m (0 to 50,000 ft)

RX26 Diskette Drive Equipment Specifications

Table A-33. RX26 Diskette Drive Description

Weight	Height	Width	Depth
425 g (0.94 lb)	25.4 mm (1.00 in)	101.6 mm (4.00 in)	150.0 mm (5.91 in)

Table A-34. RX26 Diskette Disk Drive Specifications (formatted)

Number of cylinders	80
Number of heads	2
Track density	135 tracks per inch
Diskette size	8.89 cm (3.50 in)
Recording surfaces per diskette	2
MFM data bit rate	250, 500, 1000 kHz
Sectors per track	9 double density 18 high density 36 extra density
Capacity	2.88 KB
– Bytes per drive	737,280 double density 1,474,560 high density
– Bytes per track	2,949,120 extra density
– Bytes per track	4608 double density 9216 high density 18,432 extra density
Data transfer rate	250 Kbits per second double density 500 Kbits per second high density 1000 Kbits per second extra density
Seek time	3 ms minimum 15 ms average
Average latency	100 ms
Operating power	1.25 watts (read/write) 4.60 watts (seeking)
Standby power	0.30 watts

Table A-35. RX26 Diskette Drive Operating Conditions

Temperature range ¹	°5C to 50°C (40°F to 122°F)
Temperature change rate	11°C (20°F) per hour, maximum
Relative humidity	8% to 80%, noncondensing
Maximum wet-bulb temperature	25.6°C (78°F)
Altitude	-300 to 3050 m (-1000 to 10,000 ft) maximum

¹Reduce maximum temperature by 1.8°C for each 1,000 meter (1.0°F for each 1,000 ft.) increase in altitude.

Table A-36. RX26 Diskette Drive Nonoperating Conditions

Temperature range	-40°C to 66°C (-40°F to 151°F)
Relative humidity	5% to 95%, noncondensing
Maximum wet-bulb temperature	46°C (115°F), packaged
Altitude	-300 to 12,200 m (-1000 to 40,000 ft) maximum

B

Part Numbers

This appendix lists the following part numbers for the Personal DECstation 5000:

- Basic system components
- SCSI and floppy disk hardware
- Input devices
- Cables
- Connectors, adapters, and terminators
- Miscellaneous hardware
- Software documentation
- Hardware documentation

Basic System Components

Table B-1. Basic System Components

Item	FRU Number	Customer Order Number
System module	54-21325-01	PMBA-AB
CPU module only, 20-MHz	54-20615-01	KN02-BC
CPU module only, 25-MHz	54-20615-02	KN02-CC
Power supply assembly	H7817-AA	H7817-AA
Memory module (SIMM), 2Mb, single	57-30735-02 54-21213-01*	—
Memory modules, 4Mb pair, (two 2Mb SIMMs)	—	MS01-AA MS01L-AA*
Memory module (SIMM), 8Mb, single	57-34320-01	—
Memory modules, 16Mb pair, (two 8Mb SIMMs)	—	MS01-CA
System unit cover	70-28353-02	—
System unit chassis	74-44241-02	—

¹The 54-21213-01 (MS01L-AA) memory module can be used interchangeably with the 57-30735-02 (MS01-AA) memory module and can be used with the DECstation 5000 Model 100 as well as the Personal DECstation but cannot be used with either the 2100 or 3100 models because of base system hardware incompatibility.

SCSI and Floppy Disk Hardware

Table B-2. SCSI and Floppy Disk Hardware

Item	Part Number	Customer Order Number
SCSI controller option module	54-19876-01	PMAZ-AB
SCSI system-unit-to-expansion-box cable	17-02641-01	BC09D-03
SCSI controller terminator	12-33626-01	—
SCSI chain terminator	12-30552-01	—
SCSI disk bracket kit	—	RZ2X-FN
Internal SCSI cable	17-01626-05	
RZ23L half-height hard disk drive	RZ23L-E	RZ23L-FL
RZ24L hard disk drive	RZ24L-E	RZ24L-FL
RZ24 hard disk drive PCB	29-28144-01	—
RZ25 hard disk drive	RZ25-E	RZ25-EP
RZ26 hard disk drive	RZ26-E	RZ26-EP
Internal drive power cable	17-03357-01	—
Floppy disk mounting bracket	74-45823-01	—
Floppy disk drive shield	74-46395-01	—
FDI cable	17-02221-02	—
Floppy port plug	74-46387-01	—
RX26 diskette drive	—	RX26-FC
RX26 diskette drive unit	RX26-AA	—

Monitors

Table B-3. Monitors

Item	Customer Order Number
VRC16, 16-inch color monitor, 120/240 volts, Northern Hemisphere	VRC16-HA
VRC16, 16-inch color monitor, 120/240 volts, Southern Hemisphere	VRC16-H4
VRM17, 17-inch gray-scale monitor, 120/240 volts, Northern Hemisphere	VRM17-HA
VRM17, 17-inch gray-scale monitor, 120/240 volts, Southern Hemisphere	VRM17-H4

Input Devices

Table B-4. Input Devices

Item	Customer Order Number
Keyboard, extended	LK501
Mouse	VSXXX-BB
Mouse	VSXXX-GB

Cables and Cords

Table B-5. Cables and Cords

Item	Part Number	Customer Order Number
Monitor-to-system-unit power cable (U.S.)	17-00442-25	BN19P-1K
Monitor-to-system-unit power cable (Europe)	17-00365-19	BN19A-2C
System unit or expansion box power cord	17-00606-10	BN19P-K
SCSI expansion-box-to-expansion-box cable	17-01351-04	BC19J-1E
Serial line cable	17-00811-02	BC16E-10
SCSI system-unit-to-expansion-box cable	17-02641-01	BC09D-03
ThickWire transceiver cable	17-01321-01	BNE4C-02
ThinWire cable, 12-ft	17-01241-12	BC16M-12
Video cable, color	17-02906-01	BC29G-09
Video cable, grayscale	17-02878-01	BC09M-10
Video cable, monochrome	17-03054-01	—

Miscellaneous Hardware

Table B-6. Miscellaneous Hardware

Item	FRU Number
Blank bezel insert	74-46387-01
TURBOchannel cover	74-41143-05
CPU standoff post	12-35477-01
CPU standoff rivet	12-35477-02
Disposable grounding wrist strap	12-36175-01
Light pipe	74-44244-01
Medallions	74-41856-13/-15

Connectors, Adapters, and Terminators

Table B-7. Connectors, Adapters, and Terminators

Item	FRU No.	Customer Order No.
MMJ loopback connector	12-25083-01	12-25083-01
ThickWire loopback connector	12-22196-02	12-22196-02
SCSI controller terminator	12-33626-01	—
SCSI chain terminator	12-30552-01	12-30552-01
ThinWire T-connector	12-25869-01	H8223-00
ThinWire terminators	12-26318-01	—
Clear-NVR jumper	12-14314-00	12-14314-00
Comm-line-to-MMJ adapter	12-33190-01	—
Comm modem loopback	29-24795-00	29-24795-00
ThinWire LAN assembly kit	22-00112-01	BC16T-12
DESTA ThickWire-to-ThinWire Ethernet adapter	70-22781-02	DESTA-BA
Twisted pair (10baseT) adapter	—	H3350-AA

Hardware Documentation

Table B-8. Hardware Documentation

Item	Customer Order Number
Personal DECstation 5000 User Documentation Kit	EK-PM300-DK-001
<i>Personal DECstation 5000 Basic Hardware Installation Guide</i>	EK-PM30A-IN-001
<i>Personal DECstation 5000 User's Guide</i>	EK-PM30E-RB-001
<i>Personal DECstation 5000 Reference Card</i>	EK-PM30B-RG
<i>Personal DECstation 5000 Maintenance Guide</i>	EK-PM30F-MG
<i>Personal DECstation Pocket Service Guide</i>	EK-PM30G-PG
<i>TURBOchannel Option User's Guide</i>	EK-TCOUG-TC
<i>TURBOchannel Maintenance Guide</i>	EK-TRBOC-MG
<i>TURBOchannel Extender Maintenance Guide</i>	EK-PM32X-MG

Software Documentation

Table B-9. Software Documentation

Item	Customer Order Number
ULTRIX Workstation Media (TK50) and Documentation Kit	QA-VV1AA-H5
ULTRIX Workstation Media (CD-ROM) and Documentation Kit	QA-VV1AA-H8
ULTRIX Workstation (OLD) On-line Documentation and Media Kit (on CDROM)	QA-VV1AL-H8
DECnet ULTRIX/RISC Media (TK50) and Documentation Kit	QA-YT9AA-H5
DEC C for ULTRIX/RISC (TK50)	QA-YSJAA-H5
<i>ULTRIX Guide to the Error Logger</i>	AE-ME95B-TE
<i>Technical Summary for RISC Processors</i>	AA-MM35A-TE
<i>Documentation Overview for RISC Processors</i>	AA-MM05A-TE

Table B-10. Miscellaneous TURBOchannel options

Item	FRU Number	Customer Order Number
SCSI controller option module	54-19876-01	PMAZ-AB
Thickwire ethernet option module	54-19874-01	PMAD-AA
FDDI interface module	DEFZA-AA	DEFZA-AA
TCE option module	54-20623-01	—

Table B-11. TURBOchannel Frame buffer options

Item	FRU Number	Customer Order Number
Mono frame buffer, 72Hz, (MX)	54-20609-01	PMAG-AA
Color frame buffer, 60Hz, (CX)	54-19815-01	PMAG-BA
Smart frame buffer,(HX), 1280X1024, 72Hz*, 1280X1024, 66Hz	54-21143-01	PMAGB-BA
Smart frame buffer,(HX), 1280X1024, 72Hz*, 1024X864, 60Hz	54-21143-02	PMAGB-BC
Smart frame buffer,(HX), 1280X1024, 72Hz*, 1024X768, 72Hz	54-21143-03	PMAGB-BE

Note: (* The default setting for all Smart frame buffers is 1280X1024 at 72Hz)

Table B-12. TURBOchannel Graphic accelerator options

Item	FRU Number	Customer Order Number
2D graphics accelerator, 66Hz, (PX)	54-20314-01	PMAG-CA
True color frame buffer, 66Hz, (TX)	30-35790-01	PMAG-JA
True color frame buffer, 66Hz, (TX)	30-35790-02	PMAGB-JA
True color frame buffer picture-in-picture (PIP) board	30-35788-01	
Lo 3D graphics accelerator, 66Hz, (PXG)	54-20185-01	PMAG-DA
Lo 3D graphics accelerator, 72Hz, (PXG+)	54-20185-03	PMAGB-DA
Lo 3D graphics accelerator, 66Hz, (PXG+)	54-20185-04	PMAGB-DC
Mid 3D graphics accelerator, 66Hz, (PXG)	54-20185-02	PMAG-EA
Mid 3D graphics accelerator with z-buffer, 72Hz, (PXG+)	54-20185-05	PMAGB-EA
Mid 3D graphics accelerator with z-buffer, 66Hz, (PXG+)	54-20185-06	PMAGB-EC
Hi 3D graphics accelerator, 66Hz, (PXG turbo)	54-20114-01	PMAG-FA
Hi 3D graphics accelerator, 72Hz, (PXG turbo+)	54-20114-02	PMAGB-FA
8 bit Z-buffer option for all lo and mid 3D	54-20410-AA	—
16 bit Z-buffer option for all lo and mid 3D	54-20352-AA	—
8 plane video memory for all 2Da and 3D modules	54-20116-AA	—

C

Connector Pin Assignments

This appendix lists pin assignments for the following connector:

- SCSI
- EIA RS232
- ThickWire Ethernet
- Power supply
- ISDN
- Sound
- SERIAL.bus
- Modem loopback
- Ethernet loopback

SCSI Connector Pin Assignments

Table C-1. SCSI Connector Pin Assignments

Pin	Signal	Pin	Signal
50	~ I/O	49	GND
48	~ REQ	47	GND
46	~ C/D	45	GND
44	~ SEL	43	GND
42	~ MSG	41	GND
40	~ RST	39	GND
38	~ ACK	37	GND
36	~ BSY	35	GND
34	GND	33	GND
32	~ ATN	31	GND
30	GND	29	GND
28	RSVD	27	RSVD
26	TERMPWR	25	NC
24	RSVD	23	RSVD
22	GND	21	GND
20	GND	19	GND
18	~ PARITY	17	GND
16	~ DATA<7>	15	GND
14	~ DATA<6>	13	GND
12	~ DATA<5>	11	GND
10	~ DATA<4>	9	GND
8	~ DATA<3>	7	GND
6	~ DATA<2>	5	GND
4	~ DATA<1>	3	GND
2	~ DATA<0>	1	GND

ThickWire Ethernet Connector Pin Assignments

Table C-2. ThickWire Ethernet Connector Pin Assignments

Pin	Source	Signal	Description
1			Shield
2	XCVR	ACOL+	Collision presence
3	KNO2	ATX+	Transmission
4		GND	Ground
5	XCVR	ARX+	Reception
6	XCVR	GND	Power return
7		CTL+	Control output
8		GND	Ground
9	XCVR	ACOL-	Collision presence
10	KNO2	ATX-	Transmission
11		GND	Ground
12	XCVR	ARX-	Reception
13	KNO2	+12V	Power
14		GND	Ground
15		CTL-	Control output

EIA RS232 Communications Connectors Pin Assignments

Table C-3. EIA RS232 Communications Connectors Pin Assignments

Pin	Source	Signal	CCITT ¹	EIA ²	Description
1		GND	101	AB	Protective ground
2	DTE ³	TxD	103	BA	Transmitted data
3	DCE ⁴	RxD	104	BB	Received data
4	DTE	RTS	105	CA	Request to send
5	DCE	CTS	106	CB	Clear to send
6	DCE	DSR	107	CC	Data set ready
7		GND	102	AB	Signal ground
8	DCE	CD	109	CF	Carrier detect
9					Unconnected
10					Unconnected
11					Unconnected
12	DCE	SPDMI	112	CI	Speed mode indicator
13					Unconnected
14					Unconnected
15	DCE	TxClock	114	DB	Transmit signal element timing
16					Unconnected
17	DCE	RxClock	115	DD	Receive signal element timing
18					Unconnected
19					Unconnected
20	DTE	DTR	108.2	CD	Data terminal ready
21					Unconnected

¹Comite Consultatif International Telegraphique et Telephonique, an international consultative committee that sets international communications standards.

²Electronic Industries Association.

³Data Terminal Equipment (DTE) refers to signals whose source in the Personal DECstation.

⁴Data Communications Equipment (DCE) refers to signals whose source is a peripheral, like a modem or a printer, that is attached to the system.

(continued on next page)

Table C-3 (Cont.). EIA RS232 Communications Connectors Pin Assignments

Pin	Source	Signal	CCITT ¹	EIA ²	Description
22	DCE	RI	125	CE	Ring indicator
23	DTE	DSRS	111	CH	Data signal rate select
24					Unconnected
25					Unconnected

¹Comite Consultatif International Telegraphique et Telephonique, an international consultative committee that sets international communications standards.

²Electronic Industries Association.

Power Supply Pin Assignments

Table C-4. Power Supply Pin Assignments

Pin	Signal
1	Speaker [1]
2	DCOK
3	+5.1V
4	GND
5	+5.1V
6	GND
7	+12.1V
8	Speaker [2]
9	System-OK LED
10	GND
11	+5.1V
12	GND
13	+5.1V
14	GND

ISDN Connector Pin Assignments

The external ISDN connector is an 8-wire modular jack (RJ45). Pins 1, 2, 7, and 8 are not used. A four wire cable with two twisted pairs would be connected from this jack to an ISDN NT or PAXB card for ISDN voice and data connections. Table C-5 shows a pinout for the ISDN connector.

Table C-5. ISDN Connector

Pin	Signal	Function
1		No connection
2		No connection
3	LOUT1	Line output (-)
4	LIN1	Line input (-)
5	LIN2	Line input (+)
6	LOUT2	Line output (+)
7		No connection
8		No connection

SERIAL.bus

The SERIAL.bus uses a four-pin shielded MOLEX SEMCONN or AMP SDL modular type connector. The MOLEX version will be unkeyed. The AMP version will have key “D”. This provides compatibility between the AMP and MOLEX part.

Data and power for the SERIAL.bus are transferred over low capacitance four-wire shielded cable. The four conductors are used for GND, SDA, SCL, and +12V (pins 1-4 respectively). Capacitance of the SCL and SDA conductors shall be less than 70pF/m (capacitance between one conductor and other conductors connected to shield). The S.b connector is located at the rear of the system enclosure and mounts to the base system module. The following table shows the pinout for the SERIAL.bus.

Table C-6. SERIAL.bus Pin Assignments

Pin	Signal	Function
1	GND	Ground
2	SDA	Serial data
3	SCL	Serial clock
4	+12V	+12 Volts

Sound Connector

A four-wire modular jack is provided for connection to an external telephone handset or headset. Note that a telephone should not be connected to this jack. The connection is intended for the external speaker and microphone that are contained in the headset. The headset or handset are connected to the sound jack with a four-wire cable with two twisted pairs. Table C-7 shows the pinout for the sound jack.

Table C-7. Sound Connector

Pin	Signal	Function
1	Audio In 1	Audio input
2	Audio Out 1	Audio output (+)
3	Audio Out 2	Audio output (-)
4	Audio In 2	Audio input ground

Loopback Connector Pin Assignments

Table C-8. Summary of Loopback Connectors

Function	Standard/ Unique	Part Number	Option Number
Communications loopback connector	Standard	12-15336-13	H3200
ThickWire loopback connector	Standard	12-22196-02	NA
ThinWire T-connector	Standard	12-25869-01	H8223
ThinWire terminator	Standard	12-26318-01	H8225

Table C-9. Ethernet Loopback Connector Pin Assignments, Part #12-22196-02

From Pin No.	Signal	To Pin No.	Signal	Description
P6-3	TRA+	P6-5	REC+	Through capacitor
P6-10	TRA–	P6-12	REC–	Through capacitor
P6-13	PWR	P6-6	RET	Through resistor and LED

Table C-10. Modem Loopback Connector Pin Assignments, Part #12-15336-13, Option #H3200

From Pin No.	Signal	To Pin No.	Signal
P4-2	TX2	P4-3	RX2
P4-4	RTS2	P4-5	CTS2
P4-6	DSR2	P4-20	DTR2
P4-12	SPDMI2	P4-23	DSRS2
P4-18	LLPBK2	P4-8	CI2
P4-18	LLPBK2	P4-22	RI2
P4-18	LLPBK2	P4-25	TMI2

D

Console Commands

This appendix discusses the following topics:

- The rules for entering console commands
- The terms used in this discussion of console commands
- The command format and the function of each console command
- The console command error messages

Entering Console Commands

This section describes the conventions to follow when you enter console commands.

Rules for entering console commands

You can use console commands when the system monitor displays the `>>` or `R>` prompt. When the system displays the `R>` prompt, you can use only the `boot` and `passwd` commands until you enter the console command password.

Follow these rules when you type console commands:

- Type uppercase and lowercase letters exactly as they appear in command lines. The system treats uppercase and lowercase letters as different input.
- Press Return after you type a command.
- Enter numbers as follows:
 - Enter decimal values as a string of decimal digits with no leading zeros (for example, 123).
 - Enter octal values as a string of octal digits with a leading zero (for example, 0177).
 - Enter hexadecimal values as a string of hexadecimal digits preceded by 0x (for example, 0x3ff).
- When reading or writing to memory, enter data as bytes, halfwords, or words. Because a word is 4 bytes, successive addresses referenced by a word are successive multiples of 4. For example, the address following 0x80000004 is 0x80000008. An error occurs if you specify an address that is not on a boundary for the data size you are using.
- The following key combinations have an immediate effect when the system is in console mode:
 - Ctrl-s freezes the screen display.
 - Ctrl-q releases a frozen screen display.
 - Ctrl-c aborts a command.
 - Ctrl-u erases a partially typed command line.

Conventions used in this chapter

- **Letters in boldface type**, like this, are to be typed exactly as they appear.
- *Letters in italic type*, like this, are variables that you replace with actual values.
- Arguments enclosed in square brackets ([]) are optional.
- Ellipses (...) follow an argument that can be repeated.
- A vertical bar (|) separates choices. You can think of the bar as a symbol meaning *or*.
- Parentheses enclose a group of values from which you must select one value. For example, `-(b | h | w)` means enter `-b` or `-h` or `-w`.

Some terms used in this chapter

Controller—A hardware device that directs the operation and communication between devices or other controllers. Each controller in the system has a unique controller ID number.

Script—A collection of console commands that run in a set order. Test scripts, which are collections of individual tests and may also contain other test scripts, are commonly used for troubleshooting the system.

Slot—The physical location of a module or modules.

- TURBOchannel option modules occupy slots 0 and 1.
- The base system occupies slot 3. Base system hardware includes the system module, CPU module, and memory modules. The system module contains the base system SCSI and Ethernet controllers, as well as SERIAL.bus, base system video, and ISDN.

Introduction to the Command Reference

The console commands are used to test the following hardware:

- System module
- CPU module
- Memory modules
- Ethernet controllers
- SCSI controllers
- SERIAL.bus
- Base system video controller
- ISDN
- Color frame buffer graphics modules
- 2D graphics accelerator modules
- 3D graphics modules

Console commands in the command reference following this section appear in the order in which they appear in the system console command Help menu.

For information about console commands used by TURBOchannel options not on this list, refer to the *TURBOchannel Maintenance Guide*.

Console command format summary

Here are the console commands and their formats displayed in the Help menu that appears when you enter:

```
?[cmd]
boot [[-z #] [-n] #/path [ARG...]]
cat SCRPT
cnfg [#]
d [-bhw] [-S #] RNG val
e [-bhwcdoux] [-S #] RNG
erl [-c]
go [ADR]
init [#] [-m] [ARG...]
ls [#]
passwd [-c] [-s]
printenv [EVN]
restart
script SCRPT
setenv EVN STR
sh [-belvS] [SCRPT] [ARG..]
t [-l] #/STR [ARG..]
unsetenv EVN
```

The following sections describes the console commands in detail. Note that the command descriptions do not always use the format that appears in the Help menu.

(Table D-1 lists each console command and its function.)

Table D-1. Console Command Functions

Command	Function
?	Displays a list of console commands and formats
boot	Boots the system
cat	Displays the contents of a script
cnfg	Displays system configuration information
d	Deposits data into memory
e	Examines memory contents
erl	Displays the error message log
go	Transfers control to a specific address
init	Resets the system
ls	Displays the scripts and other files that are in a module
passwd	Sets and clears the console password
printenv	Prints environment variables
restart	Attempts to restart the operating system software that is specified in the restart block
script	Creates a temporary script of console commands
setenv	Sets an environment variable
sh	Runs a script
t	Runs a test
test	Runs a comprehensive test script that checks the system hardware
unsetenv	Removes an environment variable

? Command

Use the ? command to display a list of available console commands and their formats. The ? command format is

? [*cmd*]

- To display the format for all available console commands, omit the optional *cmd* parameter.
- To display the format for a single command, replace the optional *cmd* parameter with the name of the command for which you want a command format display.

boot Command

Use the boot command to boot the system software. The boot command format is

boot [-z *seconds*] [-n] [*bootpath*] [-a] [*args...*]

- Include the optional -z *seconds* parameter to have the system wait before starting the bootstrap operation. Replace *seconds* with the number of seconds the system should wait before the bootstrap operation starts.
- Include the optional -n parameter to have the boot command load, but not execute, the specified file.
- Replace the optional *bootpath* parameter with the specification for the file you are using to boot. The file specification form depends on the type of boot device you use.

- To boot from Ethernet, use the file specification form

slot_number/protocol[/file].

Replace *slot_number* with the slot number of the Ethernet controller you are using to boot. The *protocol* parameter represents the name of the network protocol that performs the boot operation.

Replace *protocol* with either mop or tftp. The optional *file* parameter represents a specific file that you use to boot. For example, to use the protocol named mop to boot from the base system Ethernet, which uses slot number 3, type **boot 3/mop** and press Return.

- To boot from a drive, use the file specification form

slot_number/rz | tzscsi_id/file_name.

Replace *slot_number* with the SCSI controller slot number.

Use the rz | tz parameter to specify the type of drive that performs the boot operation.

Specify rz to boot from a hard disk or compact disc drive.

Specify tz to boot from a tape drive.

Replace *scsi_id* with the SCSI ID for the drive you are using to boot.

Replace *file_name* with the name of the specific file you want to boot.

For example, to boot the file named *vmunix* in multi-user mode from a hard disk drive with SCSI ID 1 that is on the SCSI bus connected to the base system SCSI controller in slot 3, type

boot 3/rz1/vmunix -a

and press Return.

To boot the file called *vmunix* from a tape drive that has SCSI ID 2 and is on the SCSI bus connected to a SCSI controller in option slot 1, type

boot 1/tz2/vmunix

and press Return.

- The **-a** argument is used to perform a multi-user boot operation. If you omit the **-a** argument, the system performs a single-user boot.
- The **-mg** argument is used when booting a multiscreen configuration. If you omit the **-mg** argument, a system with multiple graphics controllers installed will use only a single graphics controller. The system will then use the controller installed in the TURBOchannel slot with the lowest number.

To perform a multiuser boot on a multiscreen configuration type

boot 3/rz2/vmunix -a -mg

and press Return.

Important information about the boot command

- If you include no bootpath in the boot command, the system uses the boot environment variable as the string for the command.
- If you include any additional arguments, you must type the entire string in the boot command. The system ignores the boot environment variable whenever you specify any arguments in the boot command.
- If you use any spaces or tabs in the boot environment variable, you must surround the entire value with double quotation marks. For example, to set the boot environment variable to use the mop protocol to perform a multiuser boot from the base system Ethernet controller in slot 3, type

setenv boot "3/mop -a"

and press Return.

- For details about the boot command parameters for a given TURBOchannel option module, refer to the documentation for that module.

cat Command

Use the cat command to display the contents of a script. The cat command format is

cat *slot_number/script_name*

- Replace *slot_number* with the slot number of the module that has the contents you want to display.
- Replace *script_name* with the name of the script for which you want to display the contents.

For example, to display the individual self-tests contained in the mem-m test script in the base system, type

cat 3/test-mem-m

and press Return.

The following list of the individual tests that are in the test-mem-m test script then appears on the monitor:

```
>>cat 3/test-mem-m
t ${#}/mem/select
t ${#}/misc/wbpart
t ${#}/mem 0 0 55555555 -q
t ${#}/mem 1 0 55555555 -q
t ${#}/mem 2 0 55555555 -q
t ${#}/mem 3 0 55555555 -q
t ${#}/mem 4 0 55555555 -q
t ${#}/mem 5 0 55555555 -q
t ${#}/mem 6 0 55555555 -q
t ${#}/mem 7 0 55555555 -q
```

cnfg Command

Use the `cnfg` command to display hardware configuration information. The `cnfg` command format is

cnfg [*slot_number*]

- To display general system configuration information, type the `cnfg` command without the `slot_number` parameter.
- To display detailed configuration information for an individual module, replace the optional `slot_number` parameter with the slot number of the module for which you want a configuration display.

General System Configuration Displays

The following sample general system configuration display is for a system with optional Ethernet and SCSI modules installed:

```
>>cnfg
3:   KN04      DEC      V1.0a TCF0    ( 8 MB)
                                   (DV -- d=8)
                                   (Sb: devs = 2)
                                   (FDI:  0)
                                   (enet: 08-00-2b-24-34-77)
                                   (SCSI = 7)
1:   PMAD-AA   DEC      T5.2A TCF0    (enet: 08-00-2b-0f-43-31)
0:   PMAZ-AA   DEC      T5.2E TCF0    (SCSI = 7)
```

Lines that begin with 0 or 1 describe the modules, if any, that are in the option slots.

- The number that begins the line is the module slot number.
- The second term is the module name.
- The third term is the module vendor.
- The fourth term is the firmware version of the module.
- The fifth term is the type of firmware that is in the module ROM chip.

- The messages in parentheses in the rightmost column of the configuration display provide additional information about each module. The meaning of each message depends on the type of module being described.
 - For the system module, the six lines in this column describe base system hardware. The first line lists the amount of memory in the system. The second line lists the type of base video controller as well as the number of bits in the pixel depth. The third line lists the SERIAL.bus and the number of devices attached. The fourth line lists the Floppy Disk Interface and the number of devices installed. The fifth line lists the address for the base system Ethernet controller. The sixth line lists the ID number of the base system SCSI controller.
 - For TURBOchannel Ethernet controllers, the information in parentheses is the Ethernet station address.
 - For TURBOchannel SCSI controllers, the information in parentheses is the SCSI ID for the SCSI controller.
 - For graphics modules, the first part of the message in parentheses identifies the type of graphics module in the system, as follows:
 - CX indicates that the module is a color frame buffer graphics module.
 - PX indicates that the module is a 2D graphics accelerator module.
 - DX indicates that the module is a low 3D graphics module.
 - EX indicates that the module is a mid 3D graphics module.
 - FX indicates that the module is a high 3D graphics module.
 - HX indicates that the module is a smart frame buffer graphics module.
 - TX indicates that the module is a true color frame buffer graphics module.

The second part of the message has the form where # is the number of bits that make up the pixel depth. The third part of the message is displayed only when a z-buffer option is installed on a 3D graphics accelerator module. This part of the message has the form z=24.

Individual Configuration Displays

Individual configuration displays begin with a line identical to the one that describes the module in the general system configuration display.

Base system configuration displays

To obtain a base system configuration display, type

cnfg 3

and press Return.

The following is a sample configuration display for the base system configuration:

```

3:  KN04      DEC      V1.0a    TCF0    ( 8 MB)
                                         (DV -- d=8)
                                         (Sb: devs = 2)
                                         Name      Rev.      Vendor
                                         =====
                                         LK501-AA   AX1.3     DEC
                                         VSXXX-BB   AX1.2     DEC
                                         (FDI: 1)
                                         (enet: 08-00-2b-24-34-77)
                                         (SCSI = 7)
-----
DEV  PID                      VID                      REV  SCSI DEV
====  =====
rz1  RZ23                      (C) DEC                DEC    0A18  DIR
cache: I( 8 KB), D( 8 KB), S(1024 KB);  Scache line (32 bytes)
processor revision (3.0)
mem( B): a0000000:a07fffff ( 8 MB)

```

Notice that the display begins with the same information as used to describe the module in the general system configuration display discussed earlier in this section. The rest of the display shows details of the devices and memory installed in the base slot. This example shows one device.

The following list describes the information provided about the base system devices

- DEV lists the general category of the drive and its SCSI ID.
 - rz indicates that the drive is a hard disk or optical compact disc drive.
 - tz indicates that the drive is a tape drive.
 - The number at the end of the DEV entry is the drive SCSI ID.
- PID lists the product ID for some types of drives.
 - The term on the left indicates the specific drive type.
 - The term on the right indicates the product manufacturer.
- VID lists the drive vendor.
- REV lists the firmware revision number for the drive.
- SCSI DEV further describes the drive type.
 - DIR, which represents a direct access drive, appears in entries for hard disk drives.
 - SEQ, which represents a sequential access drive, appears in entries for tape drives.
 - CD-ROM appears in entries for optical compact disc drives.
 - dcache is data cache size, icache is instruction cache size.

The following list describes the information about base system memory, which appears in the last line of this display.

- Memory slot number. Base system memory is identified with a B, shown in parentheses that follow mem.

- Address range. This varies depending on which optional memory modules are installed.
- Amount of base system memory in the slot. This is always 8 megabytes.

Ethernet controller configuration displays

- To display an Ethernet controller option module configuration, type
cnfg slot_number
 Replace *slot_number* with the slot number of the Ethernet controller option module.
- To see the base system Ethernet controller configuration, display the base system configuration display, type
cnfg 3
 and press Return. The base system Ethernet controller configuration is displayed with the other base system configuration information.

The following is a sample Ethernet controller configuration display for an Ethernet controller option module in slot 1:

```
1:  PMAD-AA  DEC    V5.2a   TCF0    (enet: 08-00-2b-0c-e0-d1)
```

The Ethernet controller configuration display has the same meaning as the Ethernet controller description in the general system configuration display.

SCSI controller displays

- To display a SCSI controller option module configuration, type
cnfg slot_number
 Replace *slot_number* with the slot number of the SCSI controller option module.

- To see the base system SCSI controller configuration, type

cnfg 3

and press Return. The base system SCSI controller configuration is displayed with the other base system configuration information.

The following is a sample configuration display for a SCSI controller in slot 1 that supports two hard disk drives, one optical compact disc drive, and one tape drive:

```
1: PMAZ-AA   DEC   V5.2a   TCF0   (SCSI = 7)
-----
      DEV    PID                      VID    REV    SCSI DEV
=====
      rz0    RZ55   (C) DEC   DEC    0700    DIR
      rz1    RZ56   (C) DEC   DEC    0200    DIR
      rz4    RRD40  (c) DEC   DEC    0700    CD-ROM
      tz5                      SEQ
```

In the SCSI configuration display, the first line has the same meaning as the SCSI description in the general system configuration display. For an explanation of this first line, see “General System Configuration Displays” on page D-12.

Lines following the first line describe drives on the SCSI bus.

- DEV lists the general category of the drive and its SCSI ID.
 - rz indicates that the drive is a hard disk or optical compact disc drive.
 - tz indicates that the drive is a tape drive.
 - The number at the end is the drive SCSI ID.
- PID lists the product ID for some types of drives.
 - The term on the left indicates the specific drive type.
 - The term on the right indicates the product manufacturer.
- VID lists the drive vendor.
- REV lists the firmware revision number for the drive.

- SCSI DEV further describes the drive type.
 - DIR, which represents a direct access drive, appears in entries for hard disk drives.
 - SEQ, which represents a sequential access drive, appears in entries for tape drives.
 - CD-ROM appears in entries for optical compact disc drives.

3D graphics module configuration displays

To obtain a 3D graphics module configuration display, type

cnfg *slot_number*

Replace *slot_number* with the slot number for the 3D graphics module. Then press Return.

The following is a sample configuration display for a low 3D graphics module in option slot 1:

```
1:   PMAG-DA  DEC   V5.2a  TCF0  (DA: PXG -- D=8, z=24)
```

- The leftmost column lists the low 3D graphics module slot number.
- PMAG-DA is the part identifier for low and mid 3D graphics modules. High 3D graphics module displays list PMAG-FA as the part identifier.
- Digital is the module manufacturer.
- The fourth column lists the firmware version that each module contains.
- The fifth column lists the firmware type in each module.
- The phrase in the rightmost column indicates the 3D graphics module type and the number of VSIMMs and z-buffer modules on the base graphics module.
 - The left part of the phrase identifies the type of graphics module in the system.
DA: PXG indicates that the module is a low 3D graphics module.

EA: PXG indicates that the module is a mid 3D graphics module.

PXG_T indicates that the module is a high 3D graphics module.

- The right part of the phrase describes the VSIMM and z-buffer modules. The value after D= is the number of planes available for color generation. The value after z= is the number of bits available in the z-buffer modules.

If the last phrase in the 3D cnfg display is (ERR: invld cnfgtbl) or ends with question marks (???), there is a problem in the graphics module hardware. Refer to the *TURBOchannel Extender Maintenance Guide* to troubleshoot the 3D graphics module.

Color frame buffer graphics module configuration displays

To obtain an optional color frame buffer graphics module configuration display, type

cnfg slot_number

Replace *slot_number* with the slot number for the optional color frame buffer graphics module for which you want a configuration display and press Return.

The following is a sample color frame buffer graphics module configuration display for a color frame buffer graphics module in slot 0:

```
0:      PMAG-BA  DEC      V5.2a      TCF0      (CX -- d=8)
```

The color frame buffer graphics module configuration display has the same meaning as the color frame buffer graphics module description in the general system configuration display. For an explanation of the color frame buffer graphics module configuration display, see “General System Configuration Displays” on page D-12.

2D graphics accelerator module configuration displays

To obtain a 2D graphics accelerator module configuration display, type

cnfg *slot_number*

Replace *slot_number* with the slot number for the 2D graphics accelerator module for which you want a configuration display and press Return.

The following is a sample configuration display for a 2D graphics accelerator module in option slot 1:

```
1:    PMAG-CA  DEC    V5.2a    TCF0    (PX -- D=8)
```

The 2D graphics accelerator module display has the same meaning as the 2D graphics accelerator module description in the general configuration display. For an explanation of the 2D graphics accelerator module configuration display, see “General System Configuration Displays” on page D-12.

d Command

Use the `d` command to deposit values in memory. The `d` command format is

d [-(**b** | **h** | **w**)] [-**S** *count*] *rng*

- Use one of the optional parameters `-(b | h | w)` to specify whether to deposit the contents as bytes, halfwords, or words.
 - Specify `-b` to deposit the contents as bytes.
 - Specify `-h` to deposit the contents as halfwords.
 - Specify `-w` to deposit the contents as words.
- Include the optional `-Scount` parameter to store the same value more than once. Replace *count* with the number of times you want the value to be stored.

- Use the *rng* parameter to set the range of addresses across which the values are stored.
 - To deposit values at a single address, replace *rng* with that address.
 - To deposit a number of values across a range of addresses, replace *rng* with the address range. Use the form

address_low :address_high

 to define the range. Replace *address_low* with the starting address for storing values and replace *address_high* with the ending address for storing values.
 - To deposit values at a series of addresses, replace *rng* with the starting address and the number of successive addresses at which you want to store values. Use the form

address_low#count

 to specify the addresses where you store values. Replace *address_low* with the starting address for storing values. Replace *count* with the number of values you want to store.
 - To specify more than one address range, separate the range specifications with commas. Leave no spaces between the range specifications.

e Command

Use the `e` command to examine the contents of a specific address. The `e` command format is

`e [-b | h | w] [-c] [-d] [-o] [-u] [-x] [-S count] rng`

- Use the optional parameter (`-b | h | w`) to specify whether to examine the contents as bytes, halfwords, or words.
 - Specify `-b` to examine the contents as bytes.
 - Specify `-h` to examine the contents as halfwords.
 - Specify `-w` to examine the contents as words.
- Specify `-c` to display the data as ASCII characters.
- Specify `-d` to display the data in decimal format.
- Specify `-o` to display the contents in octal format.
- Specify `-u` to display the contents in unsigned decimal format.
- Specify `-x` to display the contents in hexadecimal format.
- Include the optional `-S count` parameter to have the command repeatedly fetch the value but display the value only once. When you type this parameter, replace *count* with the number of times you want to fetch the value.

- Use the *rng* parameter to specify the range of addresses you want to examine.
 - To examine values at a single address, replace *rng* with that address.
 - To examine values at a range of addresses, replace *rng* with the address range. Use the form

address_low :address_high

 to define the range. Replace *address_low* with the starting address for storing values and replace *address_high* with the ending address for storing values.
 - To examine values at a series of addresses, replace *rng* with the starting address and the number of successive addresses you want to examine. Use the form

address_low#count

 to specify the addresses where you store values. Replace *address_low* with the starting address for storing values. Replace *count* with the number of addresses at which you want to store values.
 - To specify more than one address range, separate each range specification with commas. Leave no spaces between the ranges.

erl Command

Use the `erl` command to display or clear the log of the errors that occurred since the most recent power-up or reset operation. The `erl` command format is

erl [-c]

- To display the current error message log, use the `erl` command without the `-c` option.
- To clear the error message log, include the `-c` option. When the error log buffer is full, no more messages are added until the buffer has been cleared by the `erl -c` command.

go Command

Use the `go` command to transfer system control to a specific system address. The `go` command format is

go [*address*]

- To transfer system control to the address specified in the last `boot -n` command, type the `go` command without the *address* parameter. If you omit the *address* parameter, and if no previous `boot -n` command has been issued, the system ignores the `go` command.
- To transfer system control to the contents of a specific address, include the *address* parameter. Replace the *address* parameter with the address to which you want to transfer control.

init Command

Use the `init` command to initialize module hardware. The `init` command format is

init [*slot_number*] [-**m**]

- To initialize the entire system, specify the `init` command with no additional arguments.
- To initialize an individual module, replace the optional *slot_number* parameter with the slot number of the module you want to initialize.
- If you perform an `init` operation on the system module (slot 3), include the optional `-m` parameter to zero all memory modules in the system module.

ls Command

Use the `ls` command to list the scripts and other objects that are in system ROM. The `ls` command format is

ls [*slot_number*]

To display a list of scripts or other objects that are available in an individual module, replace the optional *slot_number* parameter with the slot number of the module that contains the files you want to display.

The following sample display is a portion of the `ls` display for the base system in slot 2:

```
>>ls 2
      28 1 cnfg -> code
      28 1 boot -> code
      24 1 rst-q -> rst
      24 1 rst-t -> rst
      28 1 rst-m ->powerup
      32 1 test-ni-m -> test-ni-t
      28 1 init -> code
    304 1 powerup
      44 1 reset
      36 1 halt-r
      28 1 halt-b
    192 1 pst-m
    272 1 pst-q
    196 1 pst-t
      96 1 tech
    156 1 test
```

The third column of the display lists the names of the scripts and other objects in the module ROM in the specified slot.

passwd Command

Use the `passwd` command to enter, set, or clear a password. The `passwd` command format is

`passwd [-c] [-s]`

If the console prompt is `R>`, you can use only the `boot` and `passwd` commands until you enter the correct password.

To enter an existing password, type the `passwd` command without any additional parameters. At the `pwd:` prompt, enter the password. Then press Return.

After you enter the correct password, or if the system does not require a password, the system displays the console prompt `>>`. You can use all console commands whenever the console prompt is `>>`.

- To clear an existing password, include the `-c` parameter when you type the `passwd` command. First use the `passwd` command to enter the existing password. After the console prompt `>>` appears, type

`passwd -c`

and press Return. The system then removes the password requirement.

- To set a new password, include the `-s` parameter. Enter the new password at the `pwd:` prompt. When the `pwd:` prompt appears a second time, type the password again. If the two password entries match, the system sets the new value as the password.
- The password must have at least 6 and no more than 32 characters. The system is case sensitive and treats uppercase and lowercase letters as different characters.

printenv Command

Use the printenv command to display the list of environment variables. The printenv command format is

printenv [*variable*]

- To display the entire environment variable table, omit the optional *variable* parameter.
- To display an individual environment variable, replace *variable* with the name of the environment variable you want to display.

restart Command

Use the restart command to restart the system software. For the restart operation to succeed, the operating system software must have a restart block set up in memory. The restart command format is

restart

script Command

Use the script command to create a temporary set of console commands that run in an order you specify. The script command format is

script *name*

Replace *name* with the name you are giving the script. Then press Return.

After you press Return, type the commands you want to include in the script. Press Return after each command you type. Commands can be t or sh commands. Enter one command per line. When you finish typing the commands you are including, press Ctrl-d or press Return twice to complete the script.

To run the script, use the sh command described later in this appendix. When you run the script, the commands execute in the order in which you entered them when you created the script.

Note: *Only one custom script can be stored. When another script is created the first script is overwritten. The custom script is volatile and it will go away when the system is powered off.*

setenv Command

Use the `setenv` command to change an environment variable. Table D-2 lists the standard environment variables. When you change a standard environment variable (except `osconsole`, `enet`, and `#`), the system stores the new value in NVR and uses it until you use the `setenv` command to change it again or reset the NVR with the clear-NVR jumper. The `setenv` command format is

`setenv` *variable value*

When you type the `setenv` command,

- Replace *variable* with the name of the environment variable you want to set.
- Replace *value* with the new value you want to assign to the environment variable. Note that, if the new value contains blank spaces or tabs, you must use double quotation marks at the beginning and end of the value.

Table D-2 lists environment variables that appear in the environment variable display.

Table D-2. Standard Environment Variables for the Personal DECstation 5000

Environment Variable	Description
boot	Sets the default boot path. Refer to “boot Command” on page D-8 and “unsetenv Command” on page D-39.
console	<p>Selects the system console.</p> <ul style="list-style-type: none"> ■ Set the console variable to s to enable the terminal connected to the comm connector as the active console. (If no keyboard is attached to the base system SERIAL.bus connector, this terminal is enabled even if you do not set the variable to s.) ■ With the keyboard attached to the base system SERIAL.bus connector, set the console variable to a value other than s to enable the terminal connected to the graphics module in the TURBOchannel slot with the lowest number.
EPAWS	<p>Specifies how the system responds when a diagnostic test finds an error.</p> <ul style="list-style-type: none"> ■ Set the EPAWS variable to EPAWS to cause the system to pause when a diagnostic test finds an error. Press any key to cause the system to continue testing. ■ If the EPAWS variable is set to any value other than EPAWS, the system does not pause when an error occurs.
enet	Contains the address of the base system Ethernet interface. Although the environment variable display includes the enet address, you cannot set this variable. The system does it automatically.

(continued on next page)

Table D-2 (Cont.). Standard Environment Variables for the Personal DECstation 5000

Environment Variable	Description
haltaction	<p>Specifies how the system responds if it halts.</p> <ul style="list-style-type: none"> ■ Set the haltaction variable to b to cause the console to boot after the console performs the appropriate initialization and self-tests. ■ Set the haltaction variable to h to cause the console to halt and attempt no other action. ■ Set the haltaction variable to r to cause the console to restart and then attempt to boot if the restart operation fails.
more	<p>Sets how the screen scrolls lines of text.</p> <ul style="list-style-type: none"> ■ Set the more variable to 0 to have text scroll to the end before stopping. ■ Set the more variable to a number other than 0 to have scrolling pause after that number of lines has been displayed.
osconsole	<p>Contains the slot numbers of the console drivers. If a TTY driver from slot <i>x</i> serves as the console, osconsole is set to <i>x</i>. If a CRT driver from slot <i>y</i> and a kbd driver from slot <i>z</i> serve as the console, osconsole is set to <i>y,z</i>. Although the environment variable display includes the osconsole setting, you cannot set this variable. The system automatically sets the osconsole value.</p>
testaction	<p>Specifies the type of power-up self-test that the system runs.</p> <ul style="list-style-type: none"> – Specify q to run a quick test when the power-up self-test runs. – Specify t to specify a thorough test when the power-up self-test runs.

(continued on next page)

Table D-2 (Cont.). Standard Environment Variables for the Personal DECstation 5000

Environment Variable	Description
#	Specifies the slot number of the module that contains the current script. If no script is active, the system specifies the base system module, which uses slot number 3. Although the environment variable display includes the # setting, you cannot set this variable.

sh Command

Use the sh command to run a script. The sh command format is

sh [-b] [-e] [-l] [-v] [-S] [*slot_number/script*] [*arg...*]

- Include the optional -b parameter to execute the script directly, instead of through a subshell.
- Include the optional -e parameter to stop the script if an error occurs.
- Include the optional -l parameter to have the script loop until you press Ctrl-c.
- Include the optional -v parameter to echo the script to the console as the test runs.
- Include the optional -S parameter to suppress any error messages if the script is not found.
- Include the optional *slot_number/script* parameter to run a specific script.
 - Replace *slot_number* with the slot number of the module that has the script you want to run.
 - Replace *script* with the name of the script you want to run.
- The use of any additional arguments depends on the script you specified in the sh command.

If you have no optional arguments, you can also run a script by typing

slot_number/script.

- Replace *slot_number* with the slot number of the module that contains the script you want to run.
- Replace *script* with the name of the script you want to run.

For example, to run the thorough power-up self test script for a SCSI controller and drives in slot 1, type

sh 1/pst-t

and press Return.

t Command

Use the `t` command to run individual tests. The `t` command format is

t [-l] *slot_number/test_name* [arg...]

- Include the optional `-l` parameter to have the test loop until you press Ctrl-c or reset the system.
- Replace *slot_number* with the slot number of the module you want to test.
- Replace *test_name* with the name of the individual test you want to run.
- Uses for additional arguments depend on the test you are running. For an explanation of the additional arguments used in individual tests, refer to Appendix E .

To display the name and format of all individual tests for a module, type

t *slot_number*/?

Replace *slot_number* with the slot number of the module for which you want to display tests, and press Return.

test Command

Use the `test` command to run a thorough test of all system hardware. The `test` command format is

test

unsetenv Command

Use the `unsetenv` command to remove an environment variable. The `unsetenv` command removes a standard environment variable (except `osconsole`, `enet`, and `#`) during the current session only. When the system is reset, reinitialized, or powered up, the values of the standard environment variables revert to their previously set values. Table D-2 under “`setenv` command” earlier in this appendix lists the standard environment values. The `unsetenv` command format is

`unsetenv` *variable*

When you type the `unsetenv` command, replace *variable* with the name of the environment variable that you want to remove.

To clear the boot environment variable, use **`setenv boot`**, not **`unsetenv boot`**.

Note: *Environment variables stored in NVR can be only temporarily unset.*

Console Command Error Messages

Table D-3 lists the error messages console commands might return.

Table D-3. Console Command Error Messages

Error Message	Meaning
?EV:ev_name	The specified environment variable does not exist.
?EVV:value	The specified environment variable value is invalid.
?IO:slot_number/device	An I/O device has reported an error. <i>slot_number</i> represents the I/O device slot number, and <i>device</i> represents an additional message about the error.
?IO:slot_number/device	The module in the slot represented by <i>slot_number</i> does not recognize the device represented by <i>device</i> .
?PDE3:slot_number	The module in the slot represented by <i>slot_number</i> contains an early version of firmware. The ROM chip must be upgraded.
?SNF:script	The system did not find the script that was to be run.
?TXT:	The name specified in the script command is not a valid script name.
?STX:usage	A console command contained a syntax error. The usage parameter lists the correct syntax.
?STX:error	A console command contained a syntax error. The error parameter lists the incorrect portion of the command.
?TFL:slot_number/ test	A test failed. <i>slot_number</i> represents the slot number of the module that reported the error, and <i>test</i> represents the name of the failed test.
?TNF:	Test not found. The test name was probably typed incorrectly.

Base System Test Commands and Messages

This appendix describes the following diagnostic commands and messages:

- Base system Ethernet controller tests
- Base system ISDN controller tests
- Base system SCSI controller and drives tests
- Base system video controller tests
- CPU module tests
- Desktop bus (SERIAL.bus) tests
- EEPROM memory tests
- Floppy disk interface (FDI) tests
- Free-running counter tests
- Initial power-up tests
- Manufacturing test
- Memory module tests
- Real-time clock tests
- Serial communication chip (SCC) tests

Locating Individual Tests in This Appendix

This appendix lists the individual base system module tests in alphabetical order. Table E-1 lists the individual tests grouped by the function that they test.

Table E-1. Base system module tests and utilities

Test or Utility	Command
Base system Ethernet controller tests	
Collision test	t 3/ni/cllsn
Cyclic redundancy code (CRC) test	t 3/ni/crc
Display MOP counters utility	t 3/ni/ctrs
Ethernet-DMA registers test	t 3/ni/dma1
Ethernet-DMA transfer test	t 3/ni/dma2
Ethernet station address ROM (ESAR) test	t 3/ni/esar
External loopback test	t 3/ni/ext-lb
Internal loopback test	t 3/ni/int-lb
Interrupt request (IRQ) test	t 3/ni/int
Multicast test	t 3/ni/m-cst
Promiscuous mode test	t 3/ni/promisc
Registers test	t 3/ni/regs
Base system ISDN controller tests	
ISDN audio test	t 3/isdn/audio <i>[duration]</i>
ISDN baseline test	t 3/isdn/base <i>[dump]</i>
ISDN peripheral port	t 3/isdn/dma <i>[iterations]</i>

(continued on next page)

Table E-1 (Cont.). Base system module tests and utilities

Test or Utility	Command
Base system SCSI controller and drives tests	
SCSI controller test	t 3/scsi/cntl
SCSI send diagnostics test	t 3/scsi/sdiag <i>target_id</i> [d] [u] [s]
SCSI target test	t 3/scsi/target <i>target_id</i> [w] [l #]
Base system video controller tests	
ASCII font test	t 3/crt/ascii
Cursor memory test	t 3/crt/cursmem
Print E's	t 3/crt/e
Frame buffer test	t 3/crt/fb
Print H's	t 3/crt/h
Initialize video chip	t 3/crt/init
Patterns test	t 3/crt/patterns [<i>snglstp</i>]
Color palette test	t 3/crt/vdac
CPU module tests	
Cache data test	t 3/cache/data [(I D)] [<i>address</i>]
Cache fill test	t 3/cache/fill [(I D)] [<i>address</i>]
Cache isolate test	t 3/cache/isol [(I D)]
Cache reload test	t 3/cache/reload [(I D)] [<i>address</i>]
Cache segment test	t 3/cache/seg [(I D)] [<i>address</i>]
CPU-type utility	t 3/misc/cpu-type
Floating-point unit test	t 3/fpu
Translation lookaside buffer (TLB) probe test	t 3/tlb/prb
TLB registers test	t 3/tlb/reg [<i>patrn</i>]
Secondary cache test (R4000 only)	t 3/scache/data <i>parameters</i>

(continued on next page)

Table E-1 (Cont.). Base system module tests and utilities

Test or Utility	Command
Desktop bus (SERIAL.bus)	
tests	
Device capabilities test	t 3/ab/caps [<i>dev_adrs</i>]
Controller test	t 3/ab/cntl [<i>display</i>]
Counters test	t 3/ab/ctrs [<i>clear</i>]
Display next message	t 3/ab/nm [<i>dev_adrs</i>]
EEPROM memory tests	
EEPROM contents display	t 3/epm/dump
EEPROM reinitialization	t 3/epm/init
EEPROM signature data check	t 3/epm/sign
EEPROM read/write /modify	t 3/epm/write
Floppy disk interface (FDI) tests	
DOR access test	t 3/fdi/dor
Floppy controller test	t 3/fdi/exfdc
Floppy media read test	t 3/fdi/read [<i>report</i>]
Reset controller test	t 3/fdi/reset [<i>extensive</i>]
DOR access subtest	t 3/fdi/t
Write/read floppy media	t 3/fdi/wrtrd [<i>extensive</i>] [<i>report</i>]
Free-running counter tests	
Free-running counter interval	t 3/fctr/intvl
Free-running counter increment	t 3/fctr/run

(continued on next page)

Table E-1 (Cont.). Base system module tests and utilities

Test or Utility	Command
Manufacturing use only tests	
DIAGDN bit test	t 3/mfg/done
Trace tool	t 3/ab/t
Memory module tests	
Floating I/O memory test	t 3/mem/float10 [<i>addr</i>]
Memory module test	t 3/mem [<i>board</i>] [<i>thrshld</i>] [<i>pattn</i>]
RAM address select lines test	t 3/mem/select
Partial write test	t 3/misc/wbpart
Zero memory utility	t 3/mem/init
Real-time clock tests	
Nonvolatile RAM (NVR) test	t 3/rtc/nvr [<i>pattern</i>]
Real-time clock period test	t 3/rtc/period
Real-time clock register test	t 3/rtc/regs
Real-time test	t 3/rtc/time
Serial communication chip (SCC) tests	
Serial communication chip (SCC) access test	t 3/scc/access
SCC DMA test	t 3/scc/dma [(I E)] [<i>bd</i>] [<i>par</i>] [<i>bits</i>]
SCC interrupts test	t 3/scc/int
SCC I/O test	t 3/scc/io [(I E)]
SCC pins test	t 3/scc/pins [<i>loopback</i>] [<i>pause</i>]
SCC transmit and receive test	t 3/scc/tx-rx [(I E)] [<i>baud</i>] [<i>par</i>] [<i>bits</i>]

Tests

The following entries explain the commands, parameters, and error messages for each base system module test. The tests are presented in alphabetical order.

ab/caps—Desktop Bus Device Capabilities Test

This test causes attached valid Ab devices to be queried and displayed. To run the test, type

t 3/ab/caps [*dev_adrs*]

and press Return.

- Replace the *dev_adrs* parameter with a device's address to query and display that device only. Use the hh, 0xhh, or 0Xhh hexadecimal format to specify the address. The default value for *dev_adrs*, a, causes all attached valid Ab devices to be queried and displayed.

Desktop bus device capabilities test error messages

Desktop bus device capabilities test error messages have the form

?TFL: 3/ab/caps (*code*: *address*)

- ?TFL: 3/ab/caps indicates that the test reported an error.
- *code* represents a number that identifies the portion of the test that failed.
- *address* represents the address of the device returning the error.

Table E-2 lists codes used in test error messages.

Table E-2. Desktop Bus Device Capabilities Test Error Codes and Descriptions

Error Code	Description
1	CapString Request transmission error

(continued on next page)

Table E-2 (Cont.). Desktop Bus Device Capabilities Test Error Codes and Descriptions

Error Code	Description
2	Operation timed out
3	Requested data not transmitted

ab/cntl—Desktop Bus Controller Test.

This test verifies that the desktop bus controller chip is accessible and resets properly. To run the test, type

t 3/ab/ctrl [*display*]

and press Return.

The default value for the *display* parameter, n, causes the reset string not to be displayed. Replace *display* with d to display the bytes of the reset string returned by the Sb controller in numeric format. Replace *display* with a to display the bytes in both numeric and ASCII format.

ab/ctrs—Desktop Bus Counters Test

The desktop-bus counters test displays or clears counters related to desktop bus configuration events and errors, desktop bus driver events and errors, and desktop bus controller errors. Use it to further characterize known DTi errors, not to detect DTi errors. Unusually high controller counts may indicate failure modes. To run the desktop-bus counters test, type

t 3/ab/ctrs [*clear*]

and press Return.

The default value for the *clear* parameter, d, displays counters. Using c for *clear* clears the counters. The counters must be cleared to be reset; they are never reset automatically.

Desktop bus counters test error message

The desktop bus counters test error message

```
?TFL: 3/ab/ctrs (1:invld param xxxxx)
```

indicates that the parameter `xxxxx` is invalid.

ab/nm—Display Next Message Test

This test queries and displays the next Ab message from attached valid Ab devices. Query and display continues until you enter Cntrl-C at the console input device. To run the test, type

```
t 3/ab/nm [dev_adrs]
```

and press Return.

The default value for the optional `dev_adrs` parameter, `a`, causes all attached valid Ab devices to be queried and displayed.

Replace `dev_adrs` with the parameter value of a device's address to query and display that device only. Use the `hh`, `0xhh`, or `0Xhh` hexadecimal format.

ab/t—Desktop Bus Tables Verification Tool

This test performs a subset of the `ab/cntl` test. It can be used in manufacturing mode only.

cache/data—Cache Data Test

The cache data test writes data patterns to the cache and then reads them. To run the cache data test, type

```
t 3/cache/data [(I | D)] [address]
```

and press Return.

- For the optional cache parameter, specify `I` to test the instruction cache or `D` to test the data cache. `D` is the default value.

- Replace the optional *address* parameter with a specific cache address where you want the test to start. The default address is 80050000. Note that using *address* requires familiarity with the firmware specifications.

Cache data test error messages

Cache data test error messages have the form

?TFL:3 /cache/data (*code*: [*address=actual*, *sb expected*])

- ?TFL 3/cache/data indicates that the cache data test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- The optional *address=actual*, *sb expected* phrase indicates the expected and actual values in the cache.
 - *address* represents the address where the error message occurred.
 - *actual* represents the actual value at that address.
 - *expected* represents the expected value for that address.

Table E-3 lists codes used in cache data test error messages.

Table E-3. Cache Data Test Error Codes

Error Code	Description
1	Error occurred writing data pattern to cache RAM.
2	Cache parity error occurred while test was reading floating 1.
3	Error occurred when test read data pattern in cache.
4	Cache parity error occurred while test was reading floating 0.
5	Error occurred when test wrote address complement to cache RAM.
6	Cache parity data error occurred.
7	Error occurred reading address complement.
8	Cache address read caused a parity error.

cache/fill—Cache Fill Test

The cache fill test writes rotating data patterns to memory in spans that are twice the size of the cache and then reads the patterns. To run the cache fill test, type

t 3/cache/fill [(I | D)] [*address*]

and press Return.

- For the optional cache parameter, specify I to test the instruction cache or D to test the data cache. D is the default value.
- Replace the optional *address* parameter with a specific cache address where you want the test to start. The default address is 80050000. Note that using *address* requires familiarity with the firmware specifications.

Cache fill test error messages

Cache fill test error messages have the form

?TFL: 3/cache/fill (*description*)

- ?TFL 3/cache/fill indicates that the cache fill test reported an error.
- *description* represents an additional message that describes the error.

Table E-4 lists descriptions used in cache fill test error messages.

Table E-4. Cache Fill Test Error Descriptions

Error Description	Meaning
(PE)	Unexpected parity error occurred.

(continued on next page)

Table E-4 (Cont.). Cache Fill Test Error Descriptions

Error Description	Meaning
(<i>address</i> = <i>actual</i> , <i>sb expected</i>)	<p>Data pattern read reported a miscompare.</p> <ul style="list-style-type: none"> – <i>address</i> represents the address where the miscompare occurred. – <i>actual</i> represents the actual value at that address. – <i>expected</i> represents the expected value for that address.
(PE @ <i>address</i> (C)) Parity error occurred.	The <i>address</i> parameter lists the address where the error occurred.

cache/isol—Cache Isolate Test

The cache isolate test isolates data patterns to the cache and then reads and compares them. To run the cache isolate test, type

t 3/cache/isol [(I | D)]

and press Return.

For the optional cache parameter, specify I to test the instruction cache or D to test the data cache. D is the default value.

Cache isolate test error messages

Cache isolate test error messages have the form

?TFL: 3/cache/isol (code: [*address=actual*, *sb expected*])

- ?TFL 3/cache/isol indicates that the cache isolate test reported an error.
- *code* represents a number that identifies which portion of the test failed.

- The optional phrase *address= actual, sb expected* indicates the actual and expected values at the address where the error occurred.
 - *address* represents the address where the error occurred.
 - *actual* represents the actual value at that address.
 - *expected* represents the expected value at that address.

Table E-5 lists error codes used in cache isolate test error messages.

Table E-5. Cache Isolate Test Error Codes

Error Code	Description
1	Reading 00000000 pattern resulted in a cache parity error.
2	Reading 00000000 pattern resulted in a cache miss error.
3	Reading 00000000 pattern returned a data miscompare.
4	Reading 55555555 pattern resulted in a cache parity error.
5	Reading 55555555 pattern resulted a cache miss error.
6	Reading 55555555 pattern resulted in a data miscompare.
7	Reading AAAAAAAAAA pattern resulted in a cache parity error.
8	Reading AAAAAAAAAA pattern resulted in a cache miss error.
9	Reading AAAAAAAAAA pattern resulted in a data miscompare.
10	Reading data address pattern resulted in a cache parity error.
11	Reading data address pattern resulted in a parity error.
12	Reading data address pattern returned a miscompare error.

cache/reload—Cache Reload Test

The cache reload test writes rotating-parity data patterns to memory and then reads the patterns. To run the cache reload test, type

t 3/cache/reload [(I | D)] [*address*]

and press Return.

- For the optional cache parameter, specify I to test the instruction cache or D to test the data cache. D is the default value.
- Replace the optional *address* parameter with a specific cache address where you want the test to start. The default address is 80050000. Note that using *address* requires familiarity with the firmware specifications.

Cache reload test error messages

Cache reload test error messages have the form

?TFL: 3/cache/reload (*description*)

- ?TFL 3/cache/reload indicates that the cache reload test reported an error.
- *description* represents an additional message that describes the error.

Table E-6 lists descriptions used in cache reload test error messages.

Table E-6. Cache Reload Test Error Descriptions

Error Description	Meaning
(PE)	Unexpected parity error occurred.
(<i>address</i> = <i>actual</i> , sb <i>expected</i>)	Reading a data pattern reported a miscompare. <ul style="list-style-type: none">– <i>address</i> represents the address where the miscompare occurred.– <i>actual</i> represents the actual value at that address.– <i>expected</i> represents the expected value for that address.
(PE @ <i>address</i> (C))	Parity error occurred. The <i>address</i> parameter lists the address where the error occurred.

cache/seg—Cache Segment Test

The cache segment test checks individual cache segments. To run the cache segment test, type

t 3/cache/seg [(I | D)] [*address*]

and press Return.

- For the optional cache parameter, specify I to test the instruction cache or D to test the data cache. D is the default value.
- Replace the optional *address* parameter with a specific cache address where you want the test to start. The default address is 80050000. Note that using *address* requires familiarity with the firmware specifications.

Cache segment test error messages

Cache segment test error messages have the form

?TFL: 3/cache/seg (*code* : *description*)

- ?TFL 3/cache/seg indicates that the cache segment test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-7 describes error codes and descriptions used in cache segment test error messages. In all cases, the *address* value given in the description is the actual value at a given address. The correct value follows.

Table E-7. Cache Segment Test Error Codes and Descriptions

Error Code and Description	Meaning
(1: <i>address=xxxxxxxx, sb yyyyyyyy</i>)	Error occurred when the system tried to read the cache contents.
(2: <i>address=xxxxxxxx, sb yyyyyyyy</i>)	Error occurred when the system tried to read the memory contents.
(3: <i>address=xxxxxxxx, sb yyyyyyyy</i>)	Error occurred when the system performed a read and write operation on the uncached memory.
4: <i>address=xxxxxxxx, sb yyyyyyyy</i>	Cache data was inconsistent.

scache/data - Secondary Cache Test

This test run patterns through secondary cache RAM. To run the scache data test, type

t 3/scache/data

and press Return.

The secondary cache is implemented on the daughter-card as 11 16-bit wide static RAMs. This 176 (11 times 16) wire Scache interface is divided-up as follows:

- 25 Tag wires
- 07 Tag ECC wires
- 128 Data wires
- 16 Data ECC wires

Scache Error

Any error reported by the scache/data test indicate a fault in the secondary cache RAMs or in the interconnect between the R4000 and the secondary cache. Errors reported by this diagnostic could also be due to errors in reading or writing the memory; in-order to eliminate this possibility, first run the memory diagnostics.

crt/ascii—ASCII Font Test.

This test prints numerals and lowercase and uppercase characters. To run the test, type

t 3/crt/ascii

and press Return.

crt/cursmem—Cursor Memory Test

This test performs a write and read/verify of the 512 locations of the cursor memory. To run the test, type

t 3/crt/cursmem

and press Return.

Cursor memory test error messages

The cursur memory test error message

```
?TFL: 3/crt/cursmem (code: (Col_Pal_Exp = expected
Col_Pal_Act = actual))
```

indicates that a mismatch has been detected when the value of *code* is 1 or 2.

- ?TFL: 3/crt/cursmem indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *expected* represents the expected value.
- *actual* represents the value actually found.

crt/e—‘E’ Test

This test prints a screenful of ‘E’. To run the test, type

t 3/crt/e

and press Return.

crt/fb—Frame Buffer Test

The test performs a write/read test on the frame buffer. To run the test, type

t 3/crt/fb

and press Return.

Frame buffer test error messages

Frame buffer test error messages have the form

?TFL: 3/crt/fb (*code: address expected actual*)

- ?TFL: 3/crt/fb indicates the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *Address* represents the address where the mismatch occurred.
- *Expected* represents the correct value at the address.
- *Actual* represents the actual value at the address.

Table E-8 lists codes used in frame buffer test error messages.

Table E-8. Frame Buffer Test Error Codes and Descriptions

Error Code	Description
1	Data mismatch.
2	Address line stuck at fault.

crt/h—‘H’ Test

This test prints a screenful of ‘H’. To run the test, type

t 3/crt/h

and press Return.

crt/init—Chip Initialization Test

This test clears the frame buffer and initializes the Inmos G332 chip. To run the test, type

t 3/crt/init

and press Return.

crt/patterns—Screen Patterns Test

This test displays various patterns on the screen: a radix wheel, boxes, a grid, a grid with a circle, and vertical color bars. To run the test, type

t 3/crt/patterns [*snglstp*]

and press Return.

- Replace the *snglstp* parameter with 1 if you want to use a keystroke to move from pattern to patterns. The default value for *snglstp*, 0, causes the patterns to cycle automatically.

crt/vdac—Vdac/Color Palette Test

This test performs a write and read/verify to the 256 locations of the color palette. To run the test, type

t 3/crt/vdac

and press Return.

Vdac/color palette test error message

The Vdac/color palette test error message

```
?TFL: 3/crt/vdac (1: (Col_Pal_Exp = expected Col_Pal_Act  
= actual))
```

indicates that a mismatch has been detected. *expected* represents the expected value. *actual* represents the value actually found.

epm/dump—EEPROM Contents Display Test

This test displays the contents of EEPROM memory as hex bytes, 16 bytes per line, each line preceded by its starting offset within the EEPROM. To run the test, type

t 3/epm/dump

and press Return.

epm/init—EEPROM Reinitialization Test

This test reinitializes the EEPROM. It can be used in manufacture only.

epm/sign—EEPROM Signature Data Check Test

This test checks the EEPROM for valid ASCII signature data. To run the test, type

t 3/epm/sign

and press Return.

EEPROM signature data check error messages

EEPROM signature data check error messages have the form

?TFL: 3/epm/sign (*code: description*)

- ?TFL: 3/epm/sign indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.

Table E-9 lists codes used in EEPROM signature data check test error messages.

Table E-9. EEPROM Signature Data Check Test Error Descriptions

Error Code	Description
1	EEPROM not initialized
2	EEPROM checksum failure
3	EEPROM checksum failure

epm/write—EEPROM Read/Modify/Write Test

This test saves the original contents of the EEPROM, writes complementary data into the EEPROM, verifies the complemented data as written, restores the original data, and verifies it. It can be used in manufacturing mode only.

fctr/intvl—Free-running Counter Interval Test

This test enables the RTC period interrupt and verifies that the number of free-running counter increments corresponds to the programmed interval between RTC interrupts. To run the test, type

t 3/fctr/intvl

and press Return

Free-running counter interval test error messages

Free-running counter interval test error messages have the form

?TFL:3/fctr/intvl (*code:description*)

- ?TFL: 3/fctr/intvl indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.

Table E-10 lists codes used in free-running counter interval test error messages.

Table E-10. Free-running Counter Interval Test Error Descriptions

Error Code	Description
1	UIP (update-in-progress) bit set for too long.
2	Interrupt pending there when shouldn't be.
3	Timed out waiting for interrupt.
4	The free-running counter is fast (relative to real-time clock).
5	The free-running counter is slow (relative to real-time clock).

fctr/run—Free-Running Counter Increment Test

This test verifies that at least one increment of the free-running counter occurs within an arbitrary, predetermined number of instructions. To run the test, type

t 3/fctr/run

and press Return.

Free-running counter increment test error message

The free-running counter increment test error message

?TFL: 3/fctr/run (1:fctr no inc)

indicates that the free-running counter is not incrementing.

fdi/dor—Floppy DOR Register Test

This test verifies access to the DOR register (the only general read/write register) in the 82077 floppy controller chip. To run the test, type

t 3/fdi/dor

and press Return.

Floppy DOR register test error messages

Floppy DOR register test error messages have the form

?TFL:3/fdi/dor (*code:description: sb expected, is actual*)

- ?TFL: 3/fctl/intvl indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.
- *expected* represents the expected register data.
- *actual* represents the register data actually found.

Table E-11 lists codes used in floppy DOR register test error messages.

Table E-11. Floppy DOR Register Test Error Codes and Descriptions

Error Code and Description	Meaning
22:rst not clr	Controller reset condition could not be cleared.
101:DOR float 1	DOR register failed floating 1 write/read test.
101:DOR float 0	DOR register failed floating 0 write/read test.

fdi/exfdc—Exercise Floppy Disk Controller Test

This test verifies the programmability and functionality of the Floppy Disk Controller. To run the test, type

t 3/fdi/exfdc

and press Return.

Exercise floppy disk controller test error messages

Exercise floppy disk controller test error messages have the form

?TFL: 3/fdi/exfdc (*code*: *description*)

- ?TFL: 3/fdi/exfdc indicates that the exercise floppy disk controller test reported a problem.
- *code* represents a number that identifies the type of failure that occurred.
- *description* represents additional information that describes the failure.

Table E-12 lists error codes and descriptions used in exercise floppy disk controller test error messages.

Table E-12. Exercise Floppy Disk Controller Test Error Codes and Descriptions

Error Code and Description	Meaning
22:rst not clr	Controller reset could not be cleared.
23:DOR not clr	DOR was not cleared by hardware reset.
24:DSR not clr	DSR was not cleared by hardware reset.
25:DOR rst slf-clr	Controller DOR reset cleared itself.
26:no clr DOR rst	Controller DOR reset could not be cleared.
121:DOR init fail	Could not initialize controller DOR register.
122:Cfg parms not set	Controller configuration parameters were not properly set.
124:Cfg parms not reset (hardware)	Hardware reset failed to reset configuration parameters.
125:Cfg parms not reset (DOR)	DOR reset failed to reset configuration parameters.
126:Cfg parms not reset (DSR)	DSR reset failed to reset configuration parameters.

fdi/read—Floppy Media Read Test

This test attempts a one-sector read from cylinder 0, head 0, sector 1 of the inserted media. To run the test, type

t 3/fdi/read [*report*]

and press Return.

- Replace the *report* parameter with *r* to report when test failure is caused by the absence of a drive or media in the drive. The default value for *report*, *q*, does not report failures of this kind.

Floppy media read test error messages

Floppy media read test error messages have the form

?TFL: 3/fdi/read (*code:description*)

- ?TFL: 3/fdi/read indicates that the floppy media read test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.

Table E-13 lists codes used in floppy media read test error messages.

Table E-13. Floppy Media Read Test Error Descriptions

Error Code and Description	Meaning
2:inv arg	Invalid argument specified.
74:no drive	There is no floppy drive attached.
75:no media	Media missing or drive not latched.

fdi/reset—Reset Floppy Disk Interface Test

This test verifies the ability to reset the controller chip in the Floppy Disk Interface. To run the test, type

t 3/fdi/reset [*extensive*]

and press Return.

- Replace the *extensive* parameter with x to verify that aspects of controller chip functionality, such as expected register values during and after the reset occur as expected. The default value for *extensive*, d, causes a simple reset of the controller.

Reset floppy disk interface test error messages

Reset floppy disk interface test error messages have the form

?TFL: 3/fdi/reset (*code:description*)

- ?TFL: 3/fdi/reset indicates that the reset floppy disk interface test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.

Table E-14 lists codes used in reset floppy disk interface test error messages.

Table E-14. Reset Floppy Disk Interface Test Error Descriptions

Error Code and Description	Meaning
2: inv arg	Invalid argument specified.
22:rst not clr	Controller reset could not be cleared.
23:DOR not clr	DOR was not cleared by hardware reset.
24:DSR not clr	DSR was not cleared by hardware reset.
25:DOR rst slf-clr	Controller DOR reset cleared itself.
26:no clr DOR rst	Controller DOR reset could not be cleared.

fdi/t—DOR Subtest

This subtest of the fdi/dor test also verifies the presence of the FDI controller chip. To run the test, type

t 3/fdi/t

and press Return.

fdi/wrtrd—Write/Read Floppy Media Test

This test writes a block, reads a block, and compares the data. To run the test, type

t 3/fdi/wrtrd [*extensive*] [*report*]

and press Return.

- Replace the optional parameter *extensive* with x for full, sector-by-sector media coverage to be performed. With the default value, d, the test uses only the first and last sector of each track. With an extra density floppy, extensive mode takes about 18 times as long to complete.
- Replace the optional parameter *report* with r to report when test failure is caused by the absence of a drive or media in the drive. The default value for *report*, q, does not report failures of this kind.

To test a floppy disk, insert it, formatted, before you begin testing. If an RX23 drive is installed, you must set the environment variable *fdrive* to RX23. Otherwise you will produce a media absent/drive door open error even if a floppy is correctly inserted. Also, with an RX23 drive, extra density diskettes can be used only in double density format.

Write/read floppy media test error messages

Write/read floppy media test error messages have the form

?TFL: 3/fdi/wrtrd (*code:description*)

- ?TFL: 3/fdi/wrtrd indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents an additional message that describes the error.

Table E-15 lists codes used in floppy media read test error messages.

Table E-15. Write/Read Floppy Media Test Error Descriptions

Error Code and Description	Meaning
2: inv arg	Invalid argument specified.
74:no drive	There is no floppy drive attached.
151:data miscompare: <i>block</i> is <i>actual</i> , sb <i>expected</i>	Data verification failed. <i>block</i> is specified by track, head, sector, and block numbers. <i>actual</i> represents the actual value found at <i>block</i> . <i>expected</i> represents the value expected at <i>block</i> .

fpu—Floating-Point Unit (FPU) Test

The floating-point unit (FPU) test uses the FPU to perform simple arithmetic and compares the result to known values. To run the FPU test, type

t 3/fpu

and press Return.

FPU test error messages

FPU test error messages have the form

?TFL: 3/fpu (*code*)

- ?TFL 3/fpu indicates that the FPU test reported an error.
- *code* represents a number that identifies which portion of the test failed.

Table E-16 lists error codes used in FPU test error messages.

Table E-16. FPU Test Error Codes

Error Code	Meaning
1	Values did not match. Value should be 00000000.
2	Values did not match. Value should be 55555555.
3	Values did not match. Value should be AAAAAAAAAA.
4	Values did not match. Value should be FFFFFFFF.
5	Least-significant bit failed when the system was converting doubleword to word (CVT D. W.)
6	Most-significant bit failed when the system was converting doubleword to word (CVT D. W.)
7	Double miscompare occurred: $n+n \neq n$
8	Double miscompare occurred: $n+n = n$
9	Convert float-double. Value should be 55555555.
10	FPU CSR double error occurred.
11	Single miscompare occurred. $n+n \neq n$
12	Single miscompare occurred. $n+n = n$
13	Convert float-double. Value should be 55555555.
14	FPU CSR single error occurred. Value should be 00000000.
15	Single division failed. Value should be 00005555.
16	Single multiplication failed.
17	Double multiplication failed.
18	Double division failed.
19	Conversion error occurred. Pattern readback did not match.
21	FPU did not trap on overflow exception.
22	Didn't get FPU interrupt.

isdn/audio—ISDN Controller Audio Test

This test verifies that the Am79C30A ISDN controller Main Audio Processor's (MAP) tone generator can be operated, with the sound thus generated coming out of the on-board speaker. To run the test, type

t 3/isdn/audio *[duration]*

and press Return.

The default value for the *duration* parameter, *s*, produces a tone of one-second duration at about 1333 Hz. The value *l* produces a longer tone of varying frequency.

isdn/base—ISDN Baseline Test

This test verifies that the Am79C30A ISDN controller can be accessed, that it can be taken out of RESET properly, and that it is possible to write and read its internal (indirect) MUX registers. To run the test, type

t 3/isdn/base [*dump*]

and press Return.

The default value for the *dump* parameter, *n*, runs the test without displaying test data. The value *d* displays a formatted register dump of the entire controller chip.

ISDN baseline test error messages

ISDN baseline test error messages have the form

?TFL: 3/isdn/base (*code:description*)

- ?TFL: 3/isdn/base indicates that the test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the error.

Table E-17 lists codes used in test error messages.

Table E-17. ISDN Baseline Test Error Descriptions

Error Code	Description
1	SSR_ISDN_RESET not set; ssr_value = XXXX
2	MUX_MR1 not zero; mcr1 = XXXX
3	MUX_MR2 not zero; mcr2 = XXXX

(continued on next page)

Table E-17 (Cont.). ISDN Baseline Test Error Descriptions

Error Code	Description
4	MUX_MR3 not zero; mcr3 = XXXX
5	MUX_MR4 not zero; mcr4 = XXXX
6	MUX_MCR1 to be XXXX, was XXXX
7	MUX_MCR2 to be XXXX, was XXXX
8	MUX_MCR3 to be XXXX, was XXXX

isdn/dma—ISDN Controller Peripheral Port Test

The test drives the DMA pathway into and out of the Am79C30A ISDN chip. To run the test, type

t 3/isdn/dma [*iterations*]

and press Return.

The default value for the *iterations* parameter is 10. Replace *iterations* with another number to run the test another number of times.

ISDN controller peripheral port test error message

The ISDN controller peripheral port test error message

```
?TFL: 3/isdn/dma (1:ISDN DMA fail on pass N; N data, N DMA errors)
```

indicates failure on pass *N* of the test with *N* DMA errors.

mem/float10—Floating 1/0 Memory Test

The floating 1/0 memory test writes floating 1 and floating 0 across one location in RAM. To run the floating 1/0 memory test, type

t 3/mem/float10 [*addr*]

and press Return.

Replace the optional *addr* parameter with a specific address at which you want to start writing 1s. The default address is A0100000.

Floating 1/0 memory test error messages

The floating 1/0 memory test error message

```
?TFL: 3/mem/float10 (Err= N)
```

indicates that the test failed. *N* represents the number of errors the memory module reported.

mem—Memory Module Test

The memory module test performs a full pattern test on an entire memory module. To run the memory module test, type

```
t 3/mem [board] [thrshld] [pattern]
```

and press Return.

- Use the *board* parameter to specify which memory module you want to test.
 - To test one memory module pair, specify the slot number of the memory module pair you want to test. The default memory module slot number is 0.
 - To test all memory modules, specify an asterisk (*).
- Replace the optional *thrshld* parameter with the number of single-bit errors the test allows before the test fails.
- Replace the optional *pattern* parameter with a specific pattern you want to use in the test.

Memory module test error messages

There are two types of memory test error messages: parity error messages and compare error messages.

Parity error messages have the form

```
?TFL:3/mem (PER, cause=xxxxxxx, DBE=yyyyyyy, Bank n,  
side byte)
```

- ?TFL 3/mem PER indicates that the memory test reported a parity error.
- The *cause* parameter is the value in the cause register.
- The *DBE* parameter is the value at the data bus error address.

- The *Bank* parameter is the slot number of the memory module that reported the error.
- *side* indicates which module in the pair reported the error. If *side* is D16-31, the problem module is the one farthest from the power supply. If *side* is D0-15, the problem module is the one nearest the power supply.
- *byte* represents the byte in which the error occurred.

Compare error messages have the form

```
?TFL:3/mem (@ address=actual, expected)
```

- ?TFL 3/mem indicates that the memory test reported a compare error.
- The *address* parameter is the address at which the error occurred.
- *actual* represents the value at that address.
- *expected* represents the expected value at that address.

mem/init—Zero Memory Utility

The zero memory utility floods memory with zeros as fast as possible. To run the zero memory test, type

```
t 3/mem/init
```

and press Return.

The zero memory utility returns no error codes.

mem/select—RAM Select Lines Test

The RAM select lines test checks for RAM select line faults by performing a read and write operation on one location in each memory module. To run the RAM select lines test, type

```
t 3/mem/select
```

and press Return.

RAM select lines test error messages

In the RAM select lines test error message

?TFL: 3/mem/select (*address=actual*, *sb expected*)

- ?TFL: 3/mem/select indicates that the RAM select lines test reported an error.
- *address* represents the memory address where the error occurred.
- *actual* represents the actual value at the listed address.
- *expected* represents what the correct value at the listed address should be.

mfg/done—DIAGDN Bit Test

This test sets the DIAGDN bit in the system CSR. It is used in manufacturing mode only.

misc/cpu-type—CPU-Type Utility

The CPU-type utility displays a message that identifies the CPU type. To run the CPU-type utility, type

t 3/misc/cpu-type

and press Return.

CPU-type utility message

The CPU-type utility message has the form

3/misc/cpu-type's code: NDX-*type*

- *type* represents a code that indicates the type of CPU module in the system. Refer to “CPU Module” in Chapter 12 to determine the CPU type indicated by a specific *type* value.

misc/wbpart—Partial Write Test

The partial write test writes to a specific memory address and then checks whether the written values are correct. To run the partial write test, type

t 3/misc/wbpart

and press Return.

Partial write test error messages

Partial write test error messages have the form

?TFL: 3/misc/wbpart (*code*)

- ?TFL 3/misc/wbpart indicates that the partial write test reported an error.
- *code* represents a number that identifies which portion of the test failed.

Table E-18 lists error codes used in partial write test error messages.

Table E-18. Partial Write Test Error Codes

Error Code	Meaning
1	Pattern that was read showed mismatch on word access.
2	Byte 0 failed partial byte write.
3	Byte 1 failed partial byte write.
4	Byte 2 failed partial byte write.
5	Byte 3 failed partial byte write.
6	Halfword 0 failed partial halfword write.
7	Halfword 1 failed partial halfword write.

ni/cllsn—Collision Test

The collision test checks Ethernet collision detect circuitry by forcing a collision on transmission. To run the collision test, type

t 3/ni/cllsn

and press Return.

Collision test error messages

Collision test error messages have the form

?TFL: 3/ni/cllsn (*code: description*)

- ?TFL: 3/ni/cllsn indicates that the collision test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-19 lists error codes and descriptions used in collision test error messages.

Table E-19. Collision Test Error Codes and Descriptions

Error Code and Description	Meaning
3: cllsn not dtctd	Ethernet controller chip failed to detect an Ethernet collision.
4: xmt [x]	Ethernet controller chip transmission failed.
6: LANCE-init [x]	Ethernet controller chip failed to initialize.

ni/common—Common Diagnostic Utilities

The Ethernet tests use a common set of diagnostic utilities that are run by Ethernet controller tests. You cannot run these diagnostic utilities by themselves.

Common diagnostic utility error messages

Common diagnostic utility error messages have the form

?TFL: 3/ni/test_name (*code: description*)

- ?TFL: 3/ni/test_name indicates that *test_name* reported an error. (*test_name* represents the name of the test in which the diagnostic utility detected an error.)
- *code* represents a number that identifies which utility generated the error message.

- *description* represents additional information that describes the error.

Table E-20 lists error codes and descriptions used in common diagnostic utility error messages.

Table E-20. Common Diagnostic Utility Error Codes and Descriptions

Error Code and Description	Meaning
700: Invld param frmt [xxxxxx]	Parameter was not in a valid format. The <i>xxxxxx</i> value represents the parameter that had an invalid format.
901: err hltng LANCE	STOP bit did not halt the Ethernet controller chip.
902: LANCE-init timeout	Timeout occurred when the system tried to initialize the Ethernet controller chip.
903: LANCE-start timeout	Timeout occurred waiting for the Ethernet controller chip to start.
904: err initiating LANCE	Utility could not initialize the Ethernet controller chip.
905: LANCE-stop timeout	Timeout occurred in the Ethernet controller chip.
906: err initiating LANCE	I/O system failure occurred during Ethernet controller chip initialization.

ni/crc—Cyclic Redundancy Code (CRC) Test

The cyclic redundancy code (CRC) test checks the Ethernet CRC verification and bad CRC detection abilities. To run the CRC test, type

t 3/ni/crc

and press Return.

CRC test error messages

CRC test error messages have the form

?TFL: 3/ni/crc (*code: description [x]*)

- ?TFL: 3/ni/crc indicates that the CRC test reported a problem.
- *code* represents a number that identifies which portion of the test failed.

- *description* represents additional information that describes the failure.
- *x* represents a pass or fail code returned by one of the utilities that the test uses.

Table E-21 lists error codes and descriptions used in CRC test error messages.

Table E-21. CRC Test Error Codes and Descriptions

Error Code and Description	Meaning
2: LANCE-init [x]	System could not initialize the Ethernet controller chip.
3: xmt [x]	Error occurred during packet transmission.
5: fls CRC err	Ethernet chip incorrectly flagged a good CRC as bad.
6: rcv [x]	Error occurred receiving a packet.
7: LANCE-init [x]	Error occurred when the test attempted to initialize the Ethernet controller chip.
8: xmt [x]	Error occurred transmitting data packet.
10: bad CRC not dtctd	Ethernet chip did not detect a bad CRC in an incoming packet.
11: rcv [x]	Error occurred in packet receive operation.

ni/ctrs—Display MOP Counters Utility

The display MOP counters utility displays the current MOP counters for the base system Ethernet controller. To run the MOP counters utility, type

t 3/ni/ctrs

and press Return.

The display MOP counters utility produces no error messages.

ni/dma1—Ethernet-Direct Memory Access (DMA) Registers Test

The Ethernet-direct memory access (DMA) registers test checks the Ethernet-DMA control and error registers. The test then checks the ability of the system to detect a DMA error. To run the Ethernet-DMA registers test, type

t 3/ni/dma1

and press Return.

Ethernet-DMA registers test error messages

Ethernet-DMA registers test error messages have the form

?TFL: 3/ni/dma1 (*code*: *description*)

- ?TFL: 3/ni/dma1 indicates that the Ethernet-DMA registers test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-22 lists error codes and descriptions used in Ethernet-DMA registers test error messages.

Table E-22. Ethernet-DMA Registers Test Error Codes and Descriptions

Error Code and Description	Meaning
1: LDP wrt/rd [w=xxxxxxxx r=yyyyyyyy]	LDP register values matched when they should not. The w parameter is the value that was written to the LDP register. The r parameter is the value that was read from the LDP register.
2: LIOS wrt/rd [w=xxxxxxxx r=yyyyyyyy]	LANCE I/O Slot register values matched when they should not. The w parameter is the value that was written to the LANCE I/O Slot register. The r parameter is the value that was read from the LANCE I/O Slot register.

(continued on next page)

Table E-22 (Cont.). Ethernet-DMA Registers Test Error Codes and Descriptions

Error Code and Description	Meaning
3: LANCE select	LANCE I/O Slot register failed to select the LANCE.
4: LANCE deselect	LANCE I/O Slot register failed to deselect the LANCE.
5: err initiating LANCE	Ethernet controller chip initialization failed.
6: LANCE-init timeout	Timeout occurred waiting for the LANCE initialization to finish.
7: MER	Page boundary error was not recorded in the MER register.
8: SIR	LANCE memory error bit was not set in the SIR register.
9: LANCE-start timeout	Timeout occurred waiting for LANCE to start.

ni/dma2—Ethernet-Direct Memory Access (DMA) Transfer Test

The Ethernet-direct memory access (DMA) transfer test checks Ethernet DMA operation. To run the Ethernet-DMA transfer test, type

t 3/ni/dma2

and press Return.

Ethernet-DMA transfer test error messages

Ethernet-DMA transfer test error messages have the form

?TFL: 3/ni/dma2 (*code*: *description*)

- ?TFL: 3/ni/dma2 indicates that the Ethernet-DMA transfer test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-23 lists error codes and descriptions used in Ethernet-DMA transfer test error messages.

Table E-23. Ethernet-DMA Registers Test Error Codes and Descriptions

Error Code and Description	Meaning
2: LANCE-init [xxxxxxx]	Ethernet controller chip initialization failed. xxxxxxxx represents a code that describes the LANCE failure.
3: xmt [xxxxxxx] sz=yyyy ptrn=AA	Ethernet controller chip transmission failed. xxxxxxxx represents a code that describes the transmission failure. The sz parameter is the packet size. The ptrn parameter is the pattern the test tried to transmit.
4: rcv [xxxxxxx] sz=yyyy ptrn=AA	Ethernet controller chip receive operation failed. xxxxxxxx represents a code that describes the receive failure. The sz parameter is the packet size. The ptrn parameter is the pattern the test tried to receive.
8: LANCE-DMA	DMA error occurred after a packet was received.
9: LANCE-DMA	DMA error occurred when the test began.
10: LANCE-DMA	DMA error occurred after a packet was transmitted.

ni/esar—Ethernet Station Address ROM (ESAR) Test

The Ethernet station address ROM (ESAR) test checks the ESAR on the Ethernet controller. To run the ESAR test, type

t 3/ni/esar

and press Return.

ESAR test error messages

ESAR test error messages have the form

?TFL: 3/ni/esar (*code*: *description*)

- ?TFL: 3/ni/esar indicates that the ESAR test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-24 lists error codes and descriptions used in ESAR test error messages.

Table E-24. ESAR Test Error Codes and Descriptions

Error Code and Description	Meaning
2: ESAR[0-5] = 0's	First 6 bytes of the ESAR were 000000.
3: ESAR[0] = brdcst-sdrs	ESAR contained broadcast address.
5: checksum	ESAR checksum verification failed.
7: rvrs cpy !=	Reverse copy mismatch occurred.
8: frwrд cpy !=	Forward copy mismatch occurred.
9: ESAR[24-28] !=FF	Test pattern FF mismatch occurred.
10: ESAR[25-29] !=00	Test pattern 00 mismatch occurred.
11: ESAR[26-30] =55	Test pattern 55 mismatch occurred.
12: ESAR[27-31] !=AA	Test pattern AA mismatch occurred.

ni/ext-lb—External Loopback Test

The external loopback test checks the Ethernet controller and its connection to the network. If it fails, it may also indicate a bad connection between the NI controller and the network.

Before you run this test on the base system Ethernet controller, install a ThickWire loopback connector on the Ethernet controller. To run the external loopback test, type

t 3/ni/ext-lb

and press Return.

External loopback test error messages

External loopback test error messages have the form

?TFL: 3/ni/ext-lb (*code*: *description*)

- ?TFL: 3/ni/ext-lb indicates that the external loopback test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-25 lists error codes and descriptions used in external loopback test error messages.

Table E-25. External Loopback Test Error Codes and Descriptions

Error Code and Description	Meaning
1: LANCE-init [xxxxxxxx]	LANCE initialization failed. <i>xxxxxxxx</i> represents a code that describes the LANCE failure.
3: xmit [xxxxxxxx, yyyyyyyy] zzzzz	LANCE initialization failed. <i>xxxxxxxx</i> , <i>yyyyyyyy</i> represents a code that describes the LANCE failure. <i>zzzzz</i> represents a code that describes the likely cause of the failure.
4: rcv [xxxxxxxx, yyyyyyyy]	System did not receive packet. <i>xxxxxxxx</i> , <i>yyyyyyyy</i> represents a code that describes the receive failure.
6: pkt-data !=	Transmitted packet was not received.
7:	Diagnostic logic error!

ni/int—Interrupt Request (IRQ) Test

The interrupt request (IRQ) test checks whether the Ethernet controller can generate an interrupt to the R3000A chip. To run the IRQ test, type

t 3/ni/int

and press Return.

IRQ test error messages

IRQ test error messages have the form

?TFL: 3/ni/int (*code: description*)

- ?TFL: 3/ni/int indicates that the IRQ test reported a problem.
- *code* represents a number that identifies the type of failure that occurred.
- *description* represents additional information that describes the failure.

Table E-26 lists error codes and descriptions used in IRQ test error messages.

Table E-26. IRQ Test Error Codes and Descriptions

Error Code and Description	Meaning
1: int pndng	Pending interrupt was invalid.
2: init LANCE err = x	Error occurred when the system tried to initialize the Ethernet controller chip.
4: intr err xmt-stat=x	System generated no interrupt on packet transmission.

ni/int-lb—Internal Loopback Test

The internal loopback test sends and receives data packets to and from Ethernet in internal loopback mode. To run the internal loopback test, type

t 3/ni/int-lb

and press Return.

Internal loopback test error messages

Internal loopback test error messages have the form

?TFL: 3/ni/int-lb (code: description)

- ?TFL: 3/ni/int-lb indicates that the internal loopback test reported a problem.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-27 lists the error codes and error descriptions used in internal loopback test error messages.

Table E-27. Internal Loopback Test Error Codes and Descriptions

Error Code and Description	Meaning
1: rd ESAR err	The system could not access the Ethernet station address ROM.
2: LANCE-init [xxxxxxxx]	Error occurred initializing the Ethernet controller chip. <i>xxxxxxxx</i> represents a code that describes the transmission failure.
3: xmt [xxxxxxxx] sz=yyyy ptrn=zz	System did not transmit packet. <i>xxxxxxxx</i> represents a code that describes the transmission failure. The <i>sz</i> parameter is the size of the packet. The <i>ptrn</i> parameter is the pattern that was in the packet.
4: rcv [xxxxxxxx] sz=yyyy ptrn=zz	System did not receive packet. <i>xxxxxxxx</i> represents a code that describes the failure. The <i>sz</i> parameter is the size of the packet. The <i>ptrn</i> parameter is the pattern that was in the packet.
5: rcvd size=x, xptd=x	Packets received and packets sent had different sizes.
6: pkt-data !=	Data received and data sent did not match.
7	Received CRC was incorrect.
8: LANCE-IO sys err	Data looks OK, but DMA errors left.

ni/m-cst—Multicast Test

The multicast test checks the Ethernet ability to filter multicast packets. To run the multicast test, type

t 3/ni/m-cst

and press Return.

Multicast test error messages

Multicast test error messages have the form

?TFL: 3/ni/m-cst (*code*: *description*)

- ?TFL: 3/ni/m-cst indicates that the multicast test reported a problem.
- *code* represents a number that identifies which portion of the test
- *description* represents additional information that describes the failure.

Table E-28 lists the error codes and descriptions used in multicast test error messages.

Table E-28. Multicast Test Error Codes and Descriptions

Error Code and Description	Meaning
1: rd ESAR err	Error occurred reading Ethernet station address ROM.
2: LANCE-init [xxxxxxxx]	System failed to initialize the Ethernet controller chip. xxxxxxxx represents a code that describes the initialization failure.
3: xmt [xxxxxxxx]	Ethernet controller failed to send packet. xxxxxxxx represents a code that describes the transmission failure.
5: rcvd invld m-cst	Ethernet controller received a multicast packet when multicast function was disabled.

(continued on next page)

Table E-28 (Cont.). Multicast Test Error Codes and Descriptions

Error Code and Description	Meaning
6: rcv [xxxxxxxx]	Packet receive routine reported an error. <i>xxxxxxxx</i> represents a code that describes the receive error.
7: LANCE-init [xxxxxxxx]	Error occurred when the system tried to initialize the Ethernet chip. <i>xxxxxxxx</i> represents a code that describes the initialization error.
8: xmt [xxxxxxxx]	Ethernet controller failed to transmit packet. <i>xxxxxxxx</i> represents a code that describes the transmission error.
9: rcv [xxxxxxxx]	Ethernet did not receive expected packet. <i>xxxxxxxx</i> represents a code that describes the receive error.

ni/promisc—Promiscuous Mode Test

The promiscuous mode test checks that the Ethernet controller can receive packets in promiscuous mode. To run the promiscuous mode test, type

t 3/ni/promisc

and press Return.

Promiscuous mode test error messages

Promiscuous mode test error messages have the form

?TFL: 3/ni/promisc (*code: description*)

- ?TFL: 3/ni/promisc indicates that the promiscuous mode test reported a problem.
- *code* represents a number that identifies the type of failure that occurred.
- *description* represents additional information that describes the failure.

Table E-29 lists error codes and descriptions used in promiscuous mode test error messages.

Table E-29. Promiscuous Mode Test Error Codes and Descriptions

Error Code and Description	Meaning
2: LANCE-init [xxxxxxxx]	Ethernet controller initialization failed. <i>xxxxxxxx</i> represents a code that describes the initialization failure.
3: xmt [xxxxxxxx]	Packet transmission failed. <i>xxxxxxxx</i> represents a code that describes the transmission failure.
5: rcvd invld adrs	An inappropriate packet was received in nonpromiscuous mode.
6: rcv [xxxxxxxx]	Packet receive routine failed.
7: LANCE-init [xxxxxxxx]	System failed to initialize Ethernet controller in promiscuous mode. <i>xxxxxxxx</i> represents a code that describes the initialization failure.
8: xmt [xxxxxxxx]	Packet transmission failed. <i>xxxxxxxx</i> represents a code that describes the transmission failure.
9: rcv [xxxxxxxx]	Ethernet did not receive the expected packet while in promiscuous mode. <i>xxxxxxxx</i> represents a code that describes the receive failure.

ni/regs—Registers Test

The registers test performs a read and write operation on the Ethernet registers. To run the registers test, type

t 3/ni/regs

and press Return.

Registers test error messages

Registers test error messages have the form

?TFL: 3/ni/regs (*code: description*)

- ?TFL: 3/ni/regs indicates that the registers test reported a problem.
- *code* represents some number that identifies which portion of the test failed.

- *description* represents additional information that describes the failure.
 - The CSR[*n*] parameter, where *n* represents the number of a specific CSR register, indicates the actual value in the CSR register.
 - The *xpctd* parameter indicates the expected value for the same CSR register.

Table E-30 lists error codes and descriptions used in registers test error messages.

Table E-30. Registers Test Error Codes and Descriptions

Error Code and Description	Meaning
1: CSR[<i>n</i>]= <i>x</i> - <i>xpctd</i> 0	Write and read operation to Ethernet CSR[<i>n</i>] failed. The <i>n</i> represents the number of the CSR involved in the failure.
3: CSR[1]= <i>xxxx</i> - <i>xpctd</i> 0xFFFFE	Writing and reading 0xFFFFE failed on CSR 1. <i>xxxx</i> represents the actual value in CSR 1.
4: CSR[2]= <i>xxxx</i> - <i>xpctd</i> 00FF	Writing and reading 0x00FF failed on CSR 2. <i>xxxx</i> represents the actual value in CSR 2.
5: CSR[1]= <i>xxxx</i> - bit lk frn CSR[2]	Bit leak from CSR2 to CSR1 occurred. <i>xxxx</i> represents the actual value in CSR[1].
6: CSR[2]= <i>xxxx</i> - bit lk frn CSR[1]	Bit leak from CSR1 to CSR2 occurred. <i>xxxx</i> represents the actual value in CSR[2].
7: immediate write/read flr	Immediate write and read failure occurred.

ni/setup—LANCE Setup Utilities.

Several error messages are defined in the LANCE-setup utility routines. The test name displayed with them will actually differ from what is in the following documentation. The test name displayed will be that of the LANCE-test that called the utility routine.)

LANCE-setup utilities error messages

LANCE-setup utilities error messages have the form

?TFL: 3/ni/test_name (code: description)

- ?TFL: 3/ni/test_name indicates that test_name reported an error. (test_name represents the name of the test in which the diagnostic utility detected an error.)
- code represents a number that identifies which utility generated the error message.
- description represents additional information that describes the error.

Table E-31 lists error codes and descriptions used in LANCE-setup utilities error messages.

Table E-31. LANCE-setup utilities error codes and Descriptions

Error Code	Parameter and Meaning
700: xxxxxx	Listed parameter not in a valid numeric format. xxxxxx represents the parameter that had an invalid format.

rtc/nvr—Nonvolatile RAM (NVR) Test

The nonvolatile RAM (NVR) test checks the system module nonvolatile RAM. To run the NVR test, type

t 3/rtc/nvr [pattern]

and press Return.

When you type the NVR test command, replace the optional *pattern* parameter with a specific pattern that you want to use in the test. The default pattern is 55.

NVR test error messages

NVR test error messages have the form

?TFL: 3/rtc/nvr (code: address=actual, sb expected)

- ?TFL 3/rtc/nvr indicates that the NVR test read an incorrect pattern from the NVR.

- *code* represents a number that identifies which portion of the test failed.
- *address* represents the address at which the error occurred.
- *actual* represents the value at that address.
- *expected* represents the expected value at that address.

rtc/period—Real-Time Clock (RTC) Period Test

The real-time clock (RTC) period test checks the RTC periodic interrupt operation. To run the RTC period test, type

t 3/rtc/period

and press Return.

RTC period test error messages

RTC period test error messages have the form

?TFL: 3/rtc/period/ (*code*)

- ?TFL 3/rtc/period indicates that the RTC period test reported an error.
- *code* represents a number that identifies which portion of the test failed.

Table E-32 lists error codes used in RTC period test error messages.

Table E-32. RTC Period Test Error Codes

Error Code	Meaning
1	Update-in-progress (UIP) bit remained set past allotted time.
2	Real-time clock interrupt was pending when it should not have been.
3	Allowed time ran out while waiting for interrupt.

rtc/regs—Real-Time Clock Registers Test

The real-time clock registers test checks the real-time clock (RTC) registers. To run the real-time clock registers test, type

t 3/rtc/regs

and press Return.

Real-time clock registers test error messages

Real-time clock register test error messages have the form

?TFL: 3/rtc/regs (*code*: *description*)

- ?TFL 3/rtc/regs indicates that the real-time clock register test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-33 lists error codes and descriptions used in real-time clock register test error messages.

Table E-33. Real-Time Clock Register Test Error Codess

Error Code and Description	Meaning
1	UIP bit remained set past allotted time.
2: <i>register=actual</i> , <i>sb expected</i>	The test failed to write pattern correctly. The register value is the actual value in the named register followed by the expected value.

rtc/time—Real-Time Test

The real-time test checks times generated by the real-time clock against hard-coded time values. To run the real-time test, type

t 3/rtc/time

and press Return.

Real-time test error messages

Real-time test error messages have the form

?TFL: 3/rtc/time (*code*)

- ?TFL 3/rtc/time indicates that the real-time test reported an error.
- *code* represents a number that identifies which portion of the test failed.

Table E-34 lists error codes used in real-time test error messages.

Table E-34. Real-Time Test Error Codes

Error Code	Meaning
1	UIP bit remained set past allotted time.
2	Real-time clock interrupt was pending when it should not have been.
3	Allowed time ran out while waiting for interrupt.
4	Allowed time ran out while waiting for second interrupt.
5	UIP bit remained set past second allotted time.
6	Real-time clock seconds were not set to 0 on wraparound.
7	Real-time clock minutes were not set to 0.
8	Real-time clock hours were not set to 0.
9	Real-time clock day-of-the-week was not set to 1.
10	Real-time clock date was not set to 1.
12	Real-time clock month was not set to 1.
13	Real-time clock year was not set to 0.

scc/access—Serial Communication Chip (SCC) Access Test

The serial communication chip (SCC) access test checks whether the system can perform a read and write operation on the SCC. To run the SCC access test, type

t 3/scc/access

and press Return.

SCC access test error message

The SCC access test error message is

```
?TFL: 3/scc/access (1: reg-N: actual=0xXX xpctd=0xYY)
```

This message indicates that the read and write operation on the SCC failed.

- *N* represents the number of the register that failed the test.
- The actual value *xx* is the value in that register.
- The expected value *yy* is the expected value for that register.

scc/dma—Serial Communication Chip (SCC) Direct Memory Access (DMA) Test

The serial communication chip (SCC) direct memory access (DMA) test checks the ability of the serial communication and IO-ASIC chips to perform a DMA operation. To run the SCC DMA test, type

```
t 3/scc/dma [(I | E)] [bd] [par] [bits]
```

and press Return.

- For the optional loopback parameter, specify I to run an internal loopback or E to run an external loopback. I is the default value.
- Replace *bd* with the baud rate at which you want the test to run. You can specify one of the following baud rates:

```
300  
4800  
1200  
9600  
2400  
19200
```

The default baud rate is 9600.

- Replace the optional *par* parameter with a value that specifies the type of parity that the test uses.
 - Specify **none** to use no parity. **none** is the default value.
 - Specify **odd** to use odd parity.
 - Specify **even** to use even parity.
- Replace the optional *bits* parameter with a value that specifies the number of bits per characters that the test uses.
 - Specify **8** to use eight bits per character. **8** is the default value.
 - Specify **7** to use seven bits per character.
 - Specify **6** to use six bits per character.
 - Specify **5** to use five bits per character.

Caution: *If a console device is connected to the SCC line, this test may cause garbage characters to print on the screen. It may cause unpredictable behavior to other devices attached to the SCC line.*

SCC DMA test error messages

SCC DMA test error message have the form

```
?TFL: 3/scc/dma (code:N SIR_xptd= xxxxxxxx SIR=yyyyyyyyy  
SSR=zzzzzzzzz)
```

- ?TFL: 3/scc/dma indicates that the SCC DMA test reported an error.
- *code* represents a number that identifies which part of the test failed.
- The *SIR_xptd* value *xxxxxxx* is the expected value for the system interrupt register.
- The *SIR* value *yyyyyyyyy* is the actual value in the system interrupt register.
- The *SSR* value *zzzzzzzzz* is the value in the system status register.

Table E-35 discusses the codes used in SCC DMA test error messages.

Table E-35. SCC DMA Test Error Codes

Error Code	Meaning
1	SIR values are invalid.
2	Miscompare occurred during DMA read and write operation.
3	Overflow occurred in the receive buffer.
4	Interrupt signal was not sent to the system.

scc/int—Serial Communication Chip (SCC) Interrupts Test

The serial communication chip (SCC) interrupts test checks the ability of the SCC to perform internal, external, and countdown interrupts. To run the SCC interrupts test, type

t 3/scc/int

and press Return.

SCC interrupts test error messages

SCC interrupts test error messages have the form

?TFL: 3/scc/int (code: RR0=xx RR3=yy SIR=zzzzzzzz)

- ? (TFL: 3/scc/int) indicates that the SCC interrupts test reported an error.
- *code* represents a number that indicates which portion of the test reported the error.
 - If the number is an odd number, the bits to set the interrupt on were invalid.
 - If the number is an even number, the bits to set the interrupt off were invalid.
- The RR0 value *xx* is the contents of the SCC read register 0.
- The RR3 value *yy* is the contents of the SCC read register 3.
- The SIR value *zzzzzzzz* is the contents of the system interrupt register.

scc/io—Serial Communication Chip (SCC) I/O Test

The serial communication chip (SCC) I/O test checks the ability of the SCC to perform an I/O operation on the serial line. To run the SCC I/O test, type

t 3/scc/io [(I | E)]

and press Return.

When you type the SCC I/O test command,

For the optional loopback parameter, specify I to run an internal loopback or E to run an external loopback. I is the default value.

Caution: *If a console device is connected to the SCC line, this test may cause garbage characters to print out on the screen. It may cause unpredictable behavior to other devices attached to the SCC line.*

SCC I/O test error messages

SCC I/O test error messages have the form

?TFL: 3/scc/io (code: description)

- ?TFL: 3/scc/io indicates that the SCC I/O test reported an error.
- *code* represents a number that identifies which portion of the test failed.
- *description* represents additional information that describes the error.

Table E-36 lists error codes and descriptions used in SCC I/O test error messages.

Table E-36. SCC I/O Test Error Codes and Descriptions

Error Code and Description	Meaning
1: tx bfr not empty. status=xx	System could not write a single character because the transmit buffer was not empty. The status value is the contents of read register 0.
2: char not rcvd. status=x	CHAR AVAIL signal not received when the system was expecting a character. The status value is the contents of read register 0.
3: expctd=xx, rcvd=yy, status=zz	The character that was received was different than the transmitted character. The <i>xx</i> represents the transmitted value. The <i>yy</i> represents the received value. The status value is the contents of read register 0.

scc/pins—Serial Communication Chip (SCC) Pins Test

The serial communication chip (SCC) pins test checks the control pins on the communications connectors. To run the SCC pins test, type

t 3/scc/pins [*loopback*] [*pause*]

and press Return.

- Replace the optional *loopback* parameter with a value that specifies the loopback hardware that you attach to the communications connector being tested.
 - Specify 29-24795-00 if you attach a 29-24795 loopback connector. 29-24795-00 is the default type of loopback connector.
 - Specify H8571 if you attach an H8571 loopback connector.
 - Specify hm if you attach an hm loopback connector.
 - Specify H3200 if you attach an H3200 loopback connector.

- Replace the optional *pause* parameter with P to insert a 15-second pause in the test. The default value, N, runs the test with no pause.

Caution: *If a console device is connected to the SCC line, this test may cause garbage characters to print out on the screen. It may cause unpredictable behavior to other devices attached to the SCC line.*

Table E-37 lists the specific pin pairs that each loopback tests.

Table E-37. Pin Pairs Tested by Individual Loopback Connectors

Loopback connector	Pins pair tested	Meaning
29-24795	4-5	RTS to CTS
	23-6-8	SS to DSR and CD
		6-23 failure implies 6 broken.
		8-23 failure implies 8 broken.
H3200	20-22	6-23 8-23 failure implies 23 broken.
		DTR to RTI
		RTS to CTS
	4-5	DSR to DTR
H8571-A	6-20	SI to SS
	12-23	RTS to CTS
	4-5	DTR to DSR & CD
	20-6-8	6-20 failure implies 6 broken.
hm (H8571-A)	4-5	8-20 failure implies 8 broken.
		6-20 8-20 failure implies 20 broken.
		RTS to CTS

SCC pins test error messages

SCC pins test error message have the form

?TFL: 3/scc/pins (code: description)

- ?TFL: 3/scc/pins indicates that the SCC pins test reported an error.

- *description* represents additional information that describes the error.

Table E-38 lists error codes and descriptions used in SCC pins test error messages.

Table E-38. SCC Pins Test Error Codes and Descriptions

Error Code and Description	Meaning
1: (Invld) param [xxxxxx]	The number used in the test command to specify the loopback was invalid. xxxxxx represents the first two characters of the invalid value that was specified.
2: Strtup R-xx xptd=yy actl=zz pins	Test failed to generate the expected SCC status bits. The Strtup R value is the number of SCC register that contains the status bits. The xptd value is the expected status bits. The actl value is the actual status bits. pins represents the pin pairs for which the test was set up.
3:	Pins failed to respond properly. xxxxxx represents the numbers of one or more pin pairs that failed the test.

scc/tx-rx—Serial Communication Chip (SCC) Transmit and Receive Test

The serial communication chip (SCC) transmit and receive test checks the ability of the SCC to transmit and receive information. To run the SCC transmit and receive test, type

t 3/scc/tx-rx [(I | E)] [bd] [par] [bits]

and press Return.

- For the optional loopback parameter, specify I to run an internal loopback or E to run an external loopback. I is the default value.
- Replace the optional *bd* parameter with a value that specifies the baud rate at which the test runs. You can specify one of these baud rates:

300
4800
1200
9600
2400
19200

The default baud rate is 9600.

- Replace the optional *par* parameter with a value that specifies the type of parity that the test uses.
 - Specify **none** to use no parity. **none** is the default value.
 - Specify **odd** to use odd parity.
 - Specify **even** to use even parity.
- Replace the optional *bits* parameter with a value that specifies the number of bits per characters that the test uses.
 - Specify **8** to use eight bits per character. **8** is the default value.
 - Specify **7** to use seven bits per character.
 - Specify **6** to use six bits per character.
 - Specify **5** to use five bits per character.

Caution: *If a console device is connected to the SCC line, this test may cause garbage characters to print out on the screen. It may cause unpredictable behavior to other devices attached to the SCC line.*

SCC transmit and receive test error messages

SCC transmit and receive test error message have the form

?TFL: 3/scc/tx-rx (*code: description*)

- ?TFL: 3/scc/tx-rx indicates that the SCC transmit and receive test failed.
- *code* represents a number that indicates which portion of the test failed.

- *description* represents additional information that describes the error.

Table E-39 lists the error codes and descriptions used in SCC transmit and receive error messages.

Table E-39. SCC Transmit and Receive Test Error Codes and Descriptions

Error Code and Description	Meaning
1: tx bfr not empty. status=xx	System could not write a single character because the transmit buffer was not empty. The status value is the contents of SCC read register 0.
2: char not rcvd. status=xx	CHAR AVAIL signal not received when the system was expecting a character. The status value is the contents of SCC read register 0.
3: expctd=xx, rcvd=yy, status=zz	The character that was received was different from the transmitted character. The xx represents the transmitted value. The yy represents the received value. The status value is the contents of SCC read register 0.
4: Rx err. errs=xx	Receiving character in FIFO reported an error. The errs value is the associated input character FIFO error bits.

scsi/cntl—SCSI Controller Test

The SCSI controller test checks SCSI controller operation. To run the SCSI controller test, type

t 3/scsi/cntl

and press Return.

SCSI controller test error messages

SCSI controller test error messages have the form

? TFL: 3/scsi/cntl (*code*: *description*)

- ?TFL 3/scsi/cntl indicates that the SCSI controller test failed.
- *code* represents a number that indicates which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-40 lists error descriptions used in SCSI controller test error messages.

Table E-40. SCSI Controller Error Codes and Descriptions

Error Code and Description	Meaning
1: rd cnfg	Values written to and read from configuration register did not match.
2: fifo flg	First in, first out (FIFO) load and FIFO flags did not match.
3: cnt xfr	Write and read operation on TCL register reported a mismatch.
4: illg cmd	Command was illegal and did not generate an interrupt.
5: int reg	Controller cannot clear internal interrupt register.
6: rd cnfg	Mismatch occurred when reading the write/read configuration register.

scsi/sdiag—SCSI Send Diagnostics Test

The SCSI send diagnostics test runs the self-test for an individual SCSI device. You can specify whether the test alters drive parameters and includes a write operation. To run the SCSI send diagnostics test, type

t 3/scsi/sdiag *target_id* [d] [u] [s]

and press Return.

- Replace *target_id* with the SCSI ID of the device you want to test. The ID can be 0 through 7. The default ID is 0.
- Include the optional *d* and *u* parameters to specify the conditions that those parameters set for the specific drive you are testing. Note that the result of including *d* and *u* depends on the specific drive. To determine the effect of including *d* and *u*, refer to the service guide for the drive that you want to test.
- Include the optional *s* parameter to suppress the error message display.

SCSI send diagnostics test error messages

SCSI send diagnostic test error messages have the form

?TFL: 3/sdiag (*code*: *description*)

- ?TFL 3/scsi/sdiag indicates that the SCSI send diagnostics test reported an error.
- *code* represents a number that indicates which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-41 lists error codes and descriptions used in SCSI send diagnostics test error messages.

Table E-41. SCSI Send Diagnostics Test Error Error Codes and Descriptions

Error Code and Description	Meaning
1: dev ol	Test could not bring the unit on line.
2: dev ol	Test could not bring the unit on line.
3: sdiag	Device failed the send diagnostics test.

scsi/target—SCSI Target Test

The SCSI target test performs a read test on a specific SCSI device. If you include the optional write parameter, the test also performs a write test. To run the SCSI target test, type

t 3/scsi/target *target_id* [w] [l #]

and press Return.

- Replace the *target_id* parameter with the SCSI ID of the device you want to test. The ID can be 0 through 7. The default ID is 0.
- Specify the optional w parameter to include a write operation in the SCSI target test.
- Specify the optional l parameter to have the test repeat up to 9 times. If you include the l parameter, replace # with the number of times you want the test to repeat.

Caution: *This test can destroy existing data if run with the w option. The test writes over existing data at random.*

SCSI target test error messages

SCSI target test error messages have the form

?TFL: 3/scsi/target (*code*: *description*)

- ?TFL 3/scsi/target indicates that the SCSI target test reported an error.
- *code* represents a number that indicates which portion of the test failed.
- *description* represents additional information that describes the failure.

Table E-42 lists error codes and descriptions used in SCSI target test error messages.

Table E-42. SCSI Target Test Error Codes and Descriptions

Error Code and Description	Meaning
1: (dev ol) <i>N</i>	Test could not bring the device on line. <i>N</i> represents the SCSI ID of the device that could not be brought on line.
2: (tst nocomp) <i>N</i>	Command entered from the keyboard aborted the test. <i>N</i> represents the SCSI ID of the device being tested.
3: (ro dev) <i>N</i>	Test cannot perform write operation. Device is a read-only device. <i>N</i> represents the SCSI ID of the device specified in the test.
4: (dev type) <i>N</i>	Test does not test on the specified device. <i>N</i> represents the SCSI ID of the device specified in the test.
6: (rdCap) <i>N</i>	Read capacity command failed. <i>N</i> represents the SCSI ID of the device that could not be brought on line.
7: (rzWr) <i>N</i>	Write operation failed. <i>N</i> represents the SCSI ID of the device that failed.
8: (rzRd) <i>N</i>	Read operation failed. <i>N</i> represents the SCSI ID of the device that failed.
9: (cmp) <i>N</i>	Write and read values did not match. <i>N</i> represents the SCSI ID of the device involved in the miscompare.
10: (wrFlMrk) <i>N</i>	Write file mark failed. <i>N</i> represents the SCSI ID of the device being tested.
11: (tzWr) <i>N</i>	Write operation failed. <i>N</i> represents the SCSI ID of the device that failed.
12: (wrFlMrk) <i>N</i>	Write file mark failed. <i>N</i> represents the SCSI ID of the device being tested.
13: (spc) <i>N</i>	Space (-2) operation failed. <i>N</i> represents the SCSI ID of the device involved in the failure.
14: (spc) <i>N</i>	Space (1) operation failed. <i>N</i> represents the SCSI ID of the device involved in the failure.
15: (tzRd) <i>N</i>	Read operation failed. <i>N</i> represents the SCSI ID of the device being tested.
16: (cmp) <i>N</i>	Write and read values did not match. <i>N</i> represents the SCSI ID of the device involved in the miscompare.

tlb/prb—Translation Lookaside Buffer (TLB) Probe Test

The translation lookaside buffer (TLB) probe test checks whether all TLB registers respond to an address match operation. To run the TLB probe test, type

t 3/tlb/prb

and press Return.

TLB probe test error messages

The TLB probe test error message is

?TFL: 3/tlb/prb (match(0, N)=*actual*, sb *expected*)

- ?TFL: 3/tlb/prb (match(0, N)) indicates that the value at address 0 did not match the value at the address represented by *N*.
- *actual* represents the actual value found at the address represented by *N*.
- *expected* represents the expected value at the address represented by *N*.

tlb/reg—Translation Lookaside Buffer (TLB) Registers Test

The translation lookaside buffer (TLB) registers test performs a read and write operation on the TLB. To run the TLB registers test, type

t 3/tlb/reg [*patrn*]

and press Return.

Replace the optional *patrn* parameter with the pattern you want to use for the read and write operation. The default pattern is 55555555.

TLB registers test error messages

TLB registers test error messages have the form

?TFL: 3/tlb/regs (*description*)

- ?TFL 3/tlb/regs indicates that the TLB registers test reported an error.

- *description* represents additional information that describes the error.

Table E-43 lists descriptions used in TLB registers test error messages.

Table E-43. TLB Registers Test Error Descriptions

Description	Meaning
tlblo [N]= <i>actual</i> , sb <i>expected</i>	Pattern in TLB LO register was not the expected pattern. <i>N</i> represents the number of the register with the incorrect value. The actual and expected values follow.
tlbhi [N]= <i>actual</i> , sb <i>expected</i>	Pattern in TLB HI register was not the expected pattern. <i>N</i> represents the number of the register with the incorrect value. The actual and expected values in the register follow.

Diagnostic LED Codes

The system completes a series of tests and other functions whenever you turn on the system power. If the system halts at one of these functions, the diagnostic LED displays a code that indicates where in the power-up sequence the system halted.

Table E-44 lists the diagnostic LED codes and what they indicate about the power-up sequence.

Table E-44. Power-Up LED Codes

LED Code	Meaning
Two short flashes	Power is on. System firmware has begun execution. Instructions can be fetched from the system ROM. The system processor can execute instructions.
Off for less than 30 seconds	Power-up self test in progress.
Off for more than 30 seconds	Some failure that could not be isolated was detected during power-up self test
Long flashes	Test of REX-required memory failed.
One long flash followed by one short flash	<i>kbd/crt</i> selected as the firmware console devices.
One long flash followed by two short flashes	<i>tty</i> connected to the RS232 port has been selected as the firmware console device.
Steadily on	Firmware test sequence has been completed.

CPU and System Registers

This appendix describes the following registers:

- CPU cause of last exception
- CPU exception program counter
- CPU status
- CPU bad virtual address
- System support
- System interrupt
- System interrupt mask
- Memory error
- Memory size
- Error address
- MB register

This appendix describes the CPU and system registers. The CPU and system registers contain information that can be useful when troubleshooting.

There are two types of registers: R3000A CPU registers and Model 25 system registers. The system automatically displays R3000A CPU register information on the screen when exceptions occur. Use the e command in console mode to access Model 25 system registers.

CPU Registers

CPU Registers for R3000A Module

Table F-1 lists the R3000A CPU registers.

Table F-1. R3000A CPU Registers

Register	Description
Cause	Cause of last exception
EPC	Exception program counter
Status	Status register
BadVAddr	Bad virtual address (read only)

When an exception occurs, the system automatically displays R3000A register information in one of two formats.

The first format is as follows:

```
?TFL #/test-name (CUX, cause= xxxxxxxx) [KN02-BA]
?TFL #/test-name (UEX, cause= xxxxxxxx) [KN02-BA]
```

where

= Slot number of the option being tested
test-name = Test being run
cause = Cause register

The second format is as follows:

```
? PC: 0x451<vtr=nrml>
? CR: 0x810<ce=0,ip4,exc=AdEL>
? SR: 0x30080000<cu1,cu0,cm,ipl=8>
? VA: 0x0x451
? ER: i0x100003f0
? MER: 0x2000
```

where

PC = Address of the exception instruction
CR = Cause register
SR = Status register
VA = Virtual address of the exception
ER = Error address register
MER = Memory error register

Refer to Chapters 4, 5, 6, 7, and 9 for detailed troubleshooting information.

CPU Registers for R4000 Only

Table F-2. R4000 CPU Registers

Register	Description
Cause	Cause of last exception
EPC	Exception program counter
Status	Status register
BadVAddr	Bad virtual address (read only)
Error EPC	Error Exception program counter
Config	Configuration register

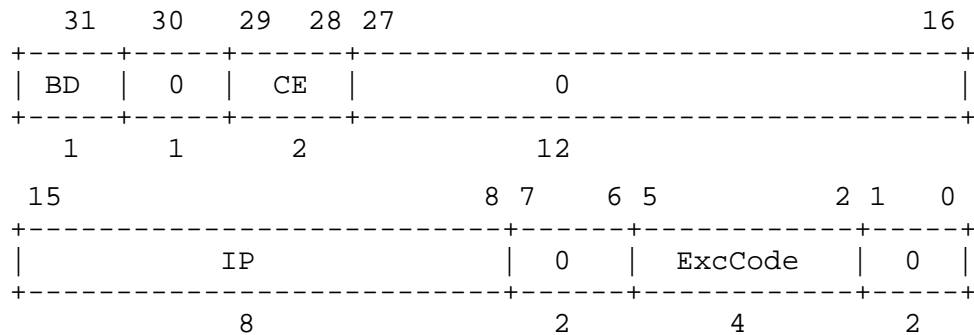
Cause Register R3000 Only

The cause register is a 32-bit read/write register that describes the nature of the last exception. A 4-bit exception code indicates the cause of the exception, and the remaining fields contain detail information relevant to the handling of certain types of exceptions.

The branch delay (BD) bit indicates whether the EPC was adjusted to point at the branch instruction that precedes the next restartable instruction. For a coprocessor unusable exception, the coprocessor error (CE) field indicates the coprocessor unit number referenced by the instructions that caused the exception.

The interrupt pending (IP) field indicates which external, internal, coprocessor, and software interrupts are pending. You can write to $IP_{1..0}$ to set or reset software interrupts. The remaining bits, $IP_{7..2}$, are read only and represent external, internal, or coprocessor interrupts.

The number and assignment of the IP bits are implementation dependent. R3000A processors have six external interrupts, where IP5 is used for the MIPS floating-point coprocessor interrupt. IP2 is normally used for system bus (I/O) interrupts. The cause register has the following format:



- BD indicates whether the last exception occurred during execution in a branch delay slot (0 = normal, 1 = delay slot).
- CE indicates the coprocessor unit number reference when a coprocessor unusable exception occurs.
- IP indicates whether an interrupt is pending.
- ExcCode is the exception code field. The exception code field is coded as shown in Table F-3.

- 0 is unused (ignored on write, zero when read).

Table F-3. Exception Codes R3000

Number	Mnemonic	Description
0	Int	Interrupt
1	Mod	TLB modification exception
2	TLBL	TLB miss exception (load or instruction fetch)
3	TLBS	TLB miss exception (store)
4	AdEL	Address error exception (load or instruction fetch)
5	AdES	Address error exception (store)
6	IBE	Bus error exception (instruction fetch)
7	DBE	Bus error exception (data reference: load or store)
8	Sys	Syscall exception
9	Bp	Breakpoint exception
10	RI	Reserved instruction exception
11	CpU	Coprocessor unusable exception
12	OV	Arithmetic overflow exception
13–31		Reserved

Cause Register R4000 Only

The R4000 Cause register for the most part is the same as the R3000 register. The R3000 uses a 4-bit exception code (ExcCode) while the R4000 uses a 5-bit exception code (ExcCode). There is also some new exception codes. Table F-4 list the exception code and a description of each code.

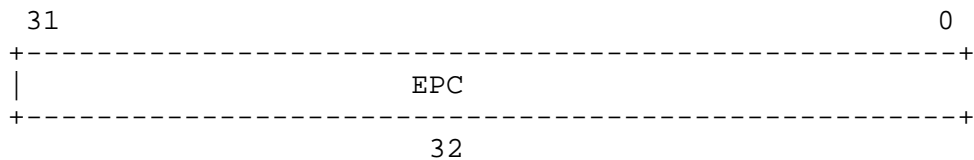
Table F-4. Exception Codes R4000

Number	Mnemonic	Description
0	Int	Interrupt
1	Mod	TLB modification exception
2	TLBL	TLB miss exception (load or instruction fetch)
3	TLBS	TLB miss exception (store)
4	AdEL	Address error exception (load or instruction fetch)
5	AdES	Address error exception (store)
6	IBE	Bus error exception (instruction fetch)
7	DBE	Bus error exception (data reference: load or store)
8	Sys	Syscall exception
9	Bp	Breakpoint exception
10	RI	Reserved instruction exception
11	CpU	Coprocessor unusable exception
12	OV	Arithmetic overflow exception
13	Tr	Trap exception
14	VCEI	Virtual Coherency Exception Instruction
15	FPE	Floating-Point exception
16-22	-	Reserved
23	WATCH	Reference to WatchHi/WatchLo address
24-30	-	Reserved
31	VCED	Virtual Coherency Exception Data

Exception Program Counter (EPC) Register R3000 and R4000

The EPC register in the R3000 and R4000 indicates the virtual address at which the most recent exception occurred. This register is a 32-bit read-only register that contains an address at which instruction processing can resume after an exception is serviced. For synchronous exceptions, the EPC register contains the virtual address of the instruction that was the direct cause of the exception. When that instruction is in a branch delay slot, the EPC register contains the virtual address of the immediately preceding branch or jump instruction.

If the exception is caused by recoverable, temporary conditions (such as a TLB miss), the EPC register contains a virtual address at the instruction that caused the exception. Thus, after correcting the conditions, the EPC registers contains a point at which execution can be legitimately resumed. The EPC register has the following format:

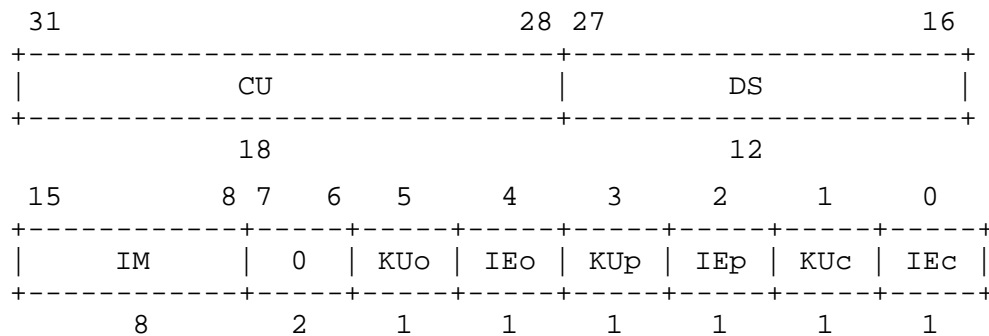


EPC is the exception program counter.

Status Register R3000 Only

The status register (SR) is a 32-bit read/write register that contains the kernel/user mode, interrupt enable, and diagnostic state of the processor. The SR contains a three-level stack (current, previous, and old) of the kernel/user (KU) bit and the interrupt enable (IE) bit. The stack is pushed when each exception is taken. The stack is popped by the restore from exception (RFE) instruction. These bits can also be directly read or written.

The status register has the following format:



- The coprocessor usability (CU) field is a 4-bit field that individually controls the usability of each of the four coprocessor unit numbers (1 = usable, 0 = unusable). Coprocessor zero is always usable in kernel mode, regardless of the setting of the CU0 bit.
- The diagnostic status (DS) field is an implementation-dependent 12-bit diagnostic status field that is used for self-testing and checking of the cache and virtual memory system. See “Diagnostic status” on page E-8 for a detailed description of the DS field.
- The interrupt mask field (IM) is an 8-bit field that controls the enabling of each of eight external interrupt conditions. It controls the enabling of each of the external, internal, coprocessor, and software interrupts (0 = disable, 1 = enable). If interrupts are enabled, an external interrupt occurs when corresponding bits are set in both the interrupt mask field of the SR and the interrupt pending (IP) field of the cause register. The actual width of this register is

machine dependent. For a description of the IP field, see “Cause Register R3000 Only” on page E-4.

- KUo is the old kernel/user mode (0 = kernel, 1 = user).
- IEo is the old interrupt enable setting (0 = disable, 1 = enable).
- KUp is the previous kernel/user mode (0 = kernel, 1 = user).
- IEp is the previous interrupt enable setting (0 = disable, 1 = enable).
- KUc is the current kernel/user mode (0 = kernel, 1 = user).
- IEc is the current interrupt enable setting (0 = disable, 1 = enable).

Diagnostic status

The diagnostic facilities depend on the characteristics of the cache and virtual memory system of the implementation. Therefore, the layout of the diagnostic status field is implementation dependent. The diagnostic status field is normally used for diagnostic code, and in certain cases, for operating system diagnostic facilities (such as reporting parity errors). On some machines it is used for relatively rare operations, such as flushing caches. Normally, this field should be set to 0 by operating system code.

The diagnostic status bits are BEV, TS, PE, CM PZ, SwC, and IsC. This set of bits provides a complete fault detection capability but is not intended to provide extensive fault diagnosis.

The SwC bit controls the switching of control signals for instructions and data caches associated with the processor. When the bit is changed, cache control signals are altered so that instruction and data caches are effectively interchanged. The processor must be executing from an uncached region and must not be executing loads or stores near the time of cache switching.

The IsC bit, when set, effectively isolates the cache currently being used as the data cache from the rest of the memory system, making cache diagnostics possible. The selection of the data cache depends on the state of the SwC bit. When the IsC bit is set, store operations affect only the cache (main memory writes are inhibited), and load operations return the data at the addressed location in the cache, whether a cache miss occurs (main memory reads are inhibited). Uncached data references are not generally useful with IsC set. Uncached store operations affect neither the cache nor the main memory system, and uncached load operations return the data at the addressed location in the cache. IsC affects only data reference; instruction fetches are not affected. This bit can also be used by operating system code to flush caches without causing associated main memory accesses.

The PZ bit, when set, causes 0 to replace the normal outgoing parity bits that are generated on a store instruction, covering both cache data and tags. This permits the writing of incorrect parity bits in the caches and the checking of each parity tree individually within the cache diagnostics.

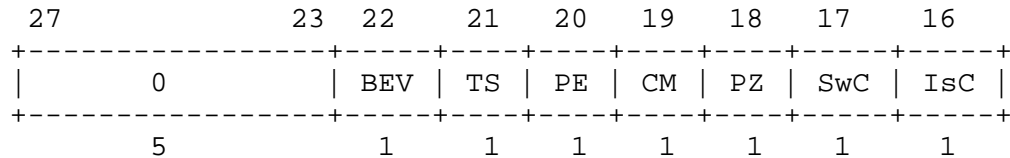
The CM bit, when the cache is isolated, indicates whether the most recent data cache load resulted in a cache miss. This bit is used by cache test programs to verify the proper functioning of the cache tag and parity bits.

The PE bit indicates whether a cache parity error occurred. It is set on a cache parity error and reset by writing 1 to this bit. Writing a 0 to this bit does not affect its value. This bit is used to log cache parity errors in software; otherwise, recovery from them is completely transparent. Within cache diagnostics, the PE bit is used to verify proper functioning of the cache parity bits and the cache parity trees.

The TS bit is read only and indicates that the TLB has shut down due to attempts to access several entries in the TLB simultaneously. This mechanism protects the TLB from hardware failures in the event of catastrophic software misuse of the TLB. When the TLB is in this state, all translations and PROBE accesses are inhibited and have undefined effects. This state can be cleared only by a reset operation.

The bootstrap exception vectors (BEV) bit, when set, relocates the UTLB miss exception vector to an address of 0xbfc00100 and the general exception vector to 0xbfc00180.

The format of the diagnostic status field is

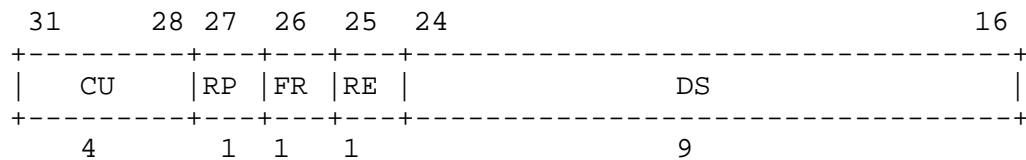


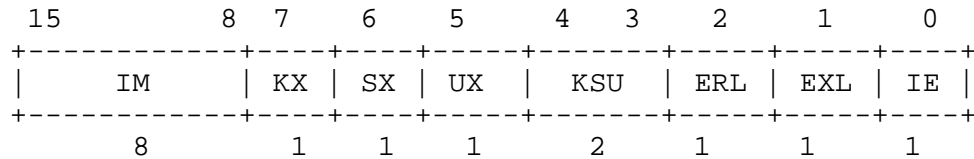
- BEV controls the location of UTLB miss and general exception vectors (0 = normal, 1 = bootstrap).
- TS indicates that TLB shut down occurred.
- PE indicates that a cache parity error occurred. This bit can be cleared by writing 1 to this bit position.
- CM indicates whether a data cache miss occurred while the system was in cache test mode (0 = hit, 1 = miss).
- PZ controls the zeroing of cache parity bits (0 = normal, 1 = parity forced to zero).
- SwC controls the switching of the data and instruction caches (0 = normal, 1 = switched).
- IsC controls isolation of the cache (0 = normal, 1 = cache isolated).
- 0 is unused (ignored on write, zero when read).

Status Register R4000 Only

The status register (SR) is a 32-bit read/write register that contains the kernel/user mode, interrupt enable, and diagnostic state of the processor. The SR contains the kernel, user, supervisor, error level, exception level, and the interrupt enable (IE) bit. These bits can also be directly read or written.

The status register has the following format:





- The (CU) coprocessor usability field is a 4-bit field that individually controls the usability of each of the four coprocessor unit numbers (1 = usable, 0 = unusable). Coprocessor zero is always usable in kernel mode, regardless of the setting of the CU0 bit.
- The (RP) reduce power enables reduced-power operation by reducing the internal clock frequency.
- The (FR) floating-point register enables additional floating-point registers (0-16 registers; 1-32 registers).
- The (RE) reverse endian when set reverse byte orientation in User mode.
- The (DS) diagnostic status field is an implementation-dependent 9-bit diagnostic status field that is used for self-testing and checking of the cache and virtual memory system. See “Diagnostic Status Field” for a detailed description of the DS field.
- The interrupt mask field (IM) is an 8-bit field that controls the eight external, interrupt, coprocessor, and software interrupts (0 = disable, 1 = enable). If interrupts are enabled, an external interrupt occurs when corresponding bits are set in both the interrupt mask field of the SR and the interrupt pending (IP) field of the cause register.
- The (KX) enables 64-bit addressing in Kernel mode.
- The (SX) enables 64-bit addressing and operations in supervisor mode.
- The (UX) enables 64-bit addressing and operations in user mode.
- The (KSU) Mode (10-User, 01_supervisor, 00-Kernel)
- The (ERL) Error level (0-normal, 1-error)
- The (EXL) Exception level (0-normal, 1-exception)
- The (IE) Interrupt Enable (0-disable, 1-enable)

Diagnostic Status Field The Diagnostic Status field has the following format:

24	23	22	21	20	19	18	17	16	
+	-	-	+	-	-	+	-	-	+
	0		BEV		TS		SR		0
+	-	-	+	-	-	+	-	-	+
	2		1		1		1		1

The following Table F-5 describes the Diagnostic Status Fields:

Table F-5. Diagnostic Status Fields

Field	Description
BEV	Controls the location of TLB refill and general exception vectors. (0 -> normal; 1 -> bootstrap)
TS	TLB shutdown has occurred (read-only)
SR	A soft reset has occurred
CH	"Hit" (tag match and valid state) or "miss" indication for last CACHE Hit Invalidate, Hit Write Back Invalidate, Hit Write Back, Hit Set Virtual, or Create Dirty Exclusion for a secondary cache.
CE	Contents of the ECC register are used to set or modify the check bits of the caches when CE equals 1.
DE	Specifies that cache parity or ECC errors are not to cause exceptions
0	Reserved. Must be written as zeroes, returns zeroes when read

BadVAddr Register R3000 and R4000

The bad virtual address (BadVAddr) register in the R3000 and R4000 is a 32-bit read-only register that contains the most recently translated virtual address for which a translation error occurred. The bad virtual address register has the following format:

BadVAddr

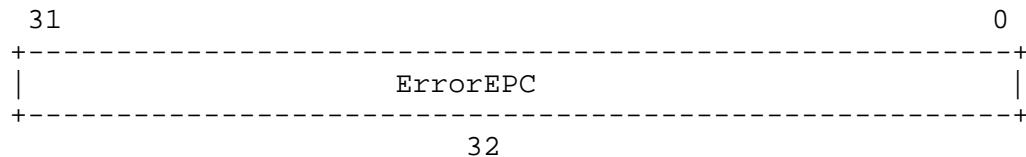
- BadVAddr is the bad virtual address.

Error Exception Program Counter (Error EPC) R4000 Only

The Error EPC register is similar to the EPC register, but it is used on ECC and parity error exceptions. It is also used to store the PC on Reset, Soft Reset and NMI exceptions. The address may be either:

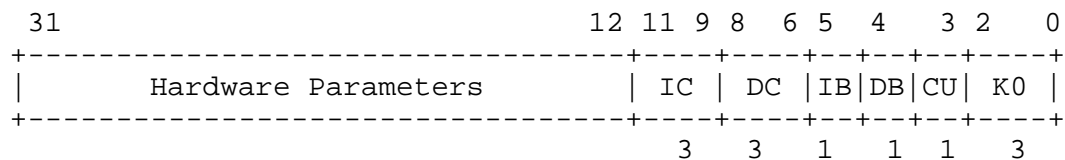
- the virtual address of the instruction that caused the exception, or
- the virtual address of the immediately preceding branch or jump instruction when that address is in a branch delay slot.

There is no branch delay slot indication for the ErrorEPC register. The ErrorEPC register has the following format.



Configuration Register R4000 Only

The Configuration Register (Config) specifies various configuration options selected on R4000 processors. Some configuration options, as defined by Config bits 31..6, are set by the hardware during reset, and are included in this register as read-only status for software. Other configuration options are read/write (defined by Config bits 5..0) and controlled by software; on reset these fields are undefined. The Config register has the following format:



Bits 31..12 indicate the setting of hardware parameters which are loaded from the serial ROM on the daughter card. For more information on these bits refer to the MIPS R4000 Microprocessor User's Manual.

Table F-6. Config Registers

Field/Bit Name	Description
IC	Primary ICache Size (ICache size = 2^{12+IC} bytes)
DC	Primary DCache Size (DCache size = 2^{12+DC} bytes)
IB	Primary ICache block size (1 -> 32 bytes; 0 -> 16 bytes)
DB	Primary DCache line size (1 -> 32 bytes; 0 -> 16 bytes)
CU	Update on Store Conditional (0 -> Store Conditional uses coherency algorithm specified by TLB; 1 -> SC uses cacheable coherent update on write)
K0	kesg0 coherency algorithm (2 -> uncached, 3 -> cached, non coherent)

System Registers

Data Buffers 3 to 0

The data buffers general-purpose 32-bit read-write registers used by the I/O control ASIC. For test purposes, these registers can be read and written. Any DMA or access to a peripheral device can overwrite these registers. To ensure proper testing, disable all DMA engines.

Table F-7 lists the system registers.

Table F-7. System Registers

Register	Console Address	Description
SSR	0xBC040100	System support register
SIR	0xBC040110	System interrupt register
Mask	0xBC040120	System interrupt mask register
MER	0xAC400000	Memory error register
MSR	0xAC800000	Memory size register
EAR	0xAE000004	Error address register

Use the `e` command to examine the contents of a model 25 system register from the console. Enter the `e` command in the following format:

e [*options*] [*console_address*]

See Appendix C, “Console Commands,” for information about the formats and options used with the `e` command and the other console commands.

System Support Register (SSR)

The SSR can be both read from and written to. Bits <31:16> are used inside the I/O control ASIC. Bits <15:0> generate signals visible outside the I/O control ASIC.

Table F-8. System Support Register 0xBC040100

Bits	Access	Description
31	R/W	Communication port transmit DMA enable (1=enable, 0=disable)
30	R/W	Communication port receive DMA enable (1=enable, 0=disable)
29	R/W	Desktop bus transmit DMA enable (1=enable, 0=disable)
28	R/W	Desktop bus receive DMA enable (1=enable, 0=disable)
27:23	R/W	Reserved
22	R/W	Floppy DMA direction, 0 = transmit (read from memory)
21	R/W	Floppy DMA enable
20	R/W	ISDN transmit DMA enable
19	R/W	ISDN receive DMA enable
18	R/W	SCSI DMA direction, 0 = transmit (read from memory)
17	R/W	SCSI DMA enable (1=enable, 0=disable)
16	R/W	LANCE DMA enable (1=enable, 0=disable)
15	R/W	DIAGDN (diagnostic flag)
14:13	R/W	Reserved
12	R/W	ISDN reset (active low)
11	R/W	SCC reset (active low)
10	R/W	RTC reset (active low)
9	R/W	53C94 SCSI controller reset (SCSI active low)
8	R/W	LANCE reset (Ethernet active low)
7	R/W	Floppy reset (active low)
6	R/W	VDAC reset (active low)
5	R/W	Desktop reset (active high)
4:1	R/W	Reserved
0	R/W	LEDs

SSR<31> This bit when set to 1 enables the communication port transmit DMA to SCC-B.

SSR<30> This bit when set to 1 enables the communication port receive DMA from SCC-B.

SSR<29> This bit when set to 1 enables the desktop bus transmit DMA.

SSR<28> This bit when set to 1 enables the desktop bus receive DMA.

SSR<27:23> Reserved, should be set to 0s.

SSR<22> Establishes the Floppy DMA direction. Setting this bit to 1 causes the DMA to move data from memory to the floppy. A 0 reverses this direction.

SSR<21> Setting this bit enables the floppy DMA. If this bit is not set to 1, floppy DMA will not advance.

SSR<20> Setting this bit enables the ISDN transmit DMA. If this bit is not set to 1, the ISDN transmit DMA will not advance.

SSR<19> Setting this bit enables the ISDN receive DMA. If this bit is not set to a 1, the ISDN receive DMA will not advance.

SSR<18> This bit, set to 0 on power up or reset, determines the direction of the SCSI DMA transfer. If the bit is a 0 then memory data will be supplied to the 53C94 upon demand from the address specified by the SCSI DMA pointer. If set to a 1, data bursts of 2 words supplied from the 53C94 are written to memory.

SSR<17> This bit when set to a 1 enables SCSI DMA; 0 disables it.

SSR<16> This bit when set to a 1 enables LANCE DMA so the Ethernet can begin functioning.

SSR<15>DIAGDN This bit is driven to the DIAGDN jumper on the base board and may be used as a trigger for debug purposes.

SSR<14:13> This bit is reserved.

SSR<12> When this bit is cleared to a 0, the 79c30 ISDN chip is reset. Because this is the power up state, this bit must be set prior to use of the ISDN or sound subsystem.

SSR<11> This signal may be read and written. The SCC UART is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the 85c30.

SSR<10> This bit may be read and written. The time-of-year controller is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the TOY. When reset the TOY loses neither its date nor its 50 bytes of permanent store.

SSR<9> This bit may be read and written. The 53C94 SCSI controller is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the 53C94.

SSR<8> This bit may be read and written. The LANCE is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the LANCE.

SSR<7> This bit may be read and written. The floppy is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the floppy controller.

SSR<6> This bit may be read and written. The video subsystem is placed in a hard reset state when this bit is 0. This bit is cleared to 0 at power up/reset, resetting the video.

SSR<5> This bit may be read and written. The desktop bus is placed in a hard reset state when this bit is 1. This bit is cleared to 0 at power up/reset. To reset the DTB this bit must be written to a 1.

SSR<4:1> These bits are reserved.

SSR<0>LED This LED is visible at the front of the machine.
LED = 1 indicates that the led is on.

System Interrupt Register (SIR)

The SIR consists of two sections. Bits <31:16> are set by the DMA engine for sundry DMA conditions. These bits are always set by the system and can be cleared by writing 0 to them. Writing 1 has no effect. These bits are cleared to 0 during system power up or reset.

Bits <15:0> reflect the status of specific system devices and are read only. A few of these are not usually used as interrupts and should be masked. These bits may or may not be reset to 0 during system power up reset, depending on the state of the interrupting device.

Table F-9. System Interrupt Register 0xBC040110

Bits	Access	Description
31	R/W0C	Communication port transmit page end interrupt
30	R/W0C	Communication port transmit DMA memory read error
29	R/W0C	Communication port receive half page interrupt
28	R/W0C	Communication port receive DMA page overrun
27	R/W0C	Desktop bus transmit page end interrupt
26	R/W0C	Desktop bus transmit DMA memory read error
25	R/W0C	Desktop bus receive half page interrupt
24	R/W0C	Desktop bus receive DMA overrun
23	R/W0C	Floppy DMA error
22	R/W0C	ISDN DMA transmit buffer pointer loaded interrupt
21	R/W0C	ISDN DMA receive buffer pointer loaded interrupt
20	R/W0C	ISDN DMA memory read or overrun error (both directions)
19	R/W0C	SCSI DMA interrupt (DMA buffer pointer loaded)
18	R/W0C	SCSI DMA overrun error
17	R/W0C	SCSI DMA memory read error
16	R/W0C	LANCE DMA memory read error
15	R	Floppy interrupt
14	R	NVR mode jumper
13	R	Power on reset
12	R	TURBOchannel option 0 interrupt
11	R	ISDN interrupt
10	R	NRMOD manufacturing mode jumper
9	R	SCSI interrupt from 53C94 SCSI controller
8	R	Ethernet interrupt
7	R	Floppy HDSTATUS
6	R	Serial interrupt
5	R	TURBOchannel option 1 interrupt
4	R	Floppy XDSTATUS
3	R	Video frame interrupt
2	R	NOT video frame interrupt

(continued on next page)

Table F-9 (Cont.). System Interrupt Register 0xBC040110

Bits	Access	Description
1	R	Desktop transmit interrupt
0	R	Desktop receive interrupt

SIR<31> This interrupt is generated by the communication port transmit DMA logic. The DMA transmitter, when enabled, transmits bytes until the pointer reaches a 4-Kbyte page boundary. At this point it stops DMA and interrupts the processor. DMA is disabled whenever this bit is set. To restart, you must clear this bit by writing a 0; writing a 1 has no effect.

SIR<30> When a parity error, page crossing error, or maximum transfer length error occurs during a communication transmit port DMA, this bit is set and the DMA is disabled. The DMA pointer will contain the error address. Check the memory sections for more information. To restart, software must clear this bit by writing a 0; writing a 1 has no effect.

SIR<29> When the receive DMA pointer associated with the communications port reaches a half page (2K) boundary, this bit is set. Software must disable DMA, then load a new pointer and restart DMA without being interrupted. This bit may be cleared by writing a 0; writing a 1 has no effect. The setting of this bit is informational only and does not stop the DMA.

SIR<28> When the receive DMA pointer associated with the communication port reaches a page boundary, this bit is set and the DMA disabled. To restart, you must clear this bit by writing a 0; writing a 1 has no effect. Note that bit <29> is set whenever this bit is set.

SIR<27> This interrupt is generated by the desktop bus transmit DMA logic. The DMA transmitter, (when enabled), transmits bytes until the pointer reaches a 4K page boundary. At this point it stops DMA and interrupts the processor. DMA is disabled whenever this bit is set. To restart, this bit must be cleared by writing a 0; writing a 1 has no effect.

SIR<26> When a parity error, page-crossing error, or maximum transfer length error occurs during a desktop bus DMA, this bit is set and the DMA is disabled. The DMA pointer will contain the error address. Check the memory sections for more information. To restart, software must clear this bit by writing a 0; writing a 1 has no effect.

SIR<25> When the receive DMA pointer associated with the desktop bus reaches a half page (2K) boundary, this bit is set. Software must disable DMA, then load a new pointer and restart DMA without being interrupted. This bit may be cleared by writing a 0; writing a 1 has no effect. The setting of this bit is informational only and does not stop the DMA.

SIR<24> When the receive DMA pointer associated with the Desktop Bus reaches a page boundary, this bit is set and the DMA disabled. To restart, you must clear this bit by writing a 0; writing a 1 has no effect. Note that bit <25> is set whenever this bit is set.

SIR<23> When a parity error, page crossing error, or maximum transfer length error occurs during a floppy DMA, this bit is set and the DMA is disabled. The DMA pointer will contain the error address. Check the memory sections for more information. To restart, software must clear this bit by writing a 0; writing a 1 has no effect.

SIR22> This bit is set when the ISDN transmit buffer pointer is loaded into the ISDN transmit pointer at the end of a page. The software will have to refill the buffer pointer or the DMA will halt when the last page is transmitted.

SIR<21> This bit is set when the ISDN receive buffer pointer is loaded into the ISDN receive pointer at the end of a page. The software will have to refill the buffer pointer or the DMA will halt when the last page is filled.

SIR<20> This bit is set when the ISDN transmit or receive DMA detects a memory parity error or a page-crossing error, or if either pointer is exhausted and the buffer pointer is empty. When this bit is set, both transmit and receive DMAs are halted. Software must examine both pointers and the MER to determine the actual cause of the error. In some cases, this may not be possible. Either the ISDN receive pointer or the transmit pointer will contain the address in error. The MER will have further error information. The software will have to write this bit to a 1 before restarting the DMA or no progress will be made.

SIR<19> This interrupt is set whenever the SCSI DMA buffer pointer associated with the SCSI port is loaded into the SCSI DMA pointer register. Software will use this interrupt to load a new buffer pointer into the SCSI buffer pointer register. This interrupt is cleared by writing a 0 to it.

SIR<18> This bit is set when the SCSI buffer pointer is not reloaded soon enough. It indicates an overrun condition as the data buffer space is exhausted. DMA is disabled when this bit is set. It may be cleared by writing a 0 to it.

SIR<17> This bit is set when the SCSI DMA encounters a memory read error while doing DMA. DMA is disabled when this bit is set. It may be cleared by writing a 0 to it.

SIR<16> This bit is set to 1 when the LANCE DMA encounters a memory read error, disabling DMA. The LANCE will then time out, interrupting the processor to handle the problem. The address of the error will be visible in the LPR. It may be cleared by writing a 0 to it; writing 1s to it has no effect.

SIR<15> This bit is set to 1 when the floppy controller generates an interrupt.

SIR<14>UNSCUR When an external jumper is installed, this bit is set and the contents of the NV RAM in the TOY are set to default system values by software. Any password that had been saved is gone.

SIR<13> This bit is not set after a power-on reset. It will be set after a reset from the desktop bus.

SIR<12> This bit reflects the state of the TURBOchannel slot 0 interrupt. A 1 indicates an interrupting condition.

SIR<11> ISDN interrupt.

SIR<10>NRMOD This bit reflects the state of the manufacturing jumper on the module. When the jumper is absent, NRMMOD is 0 and the console should perform its normal power-up/reset tests and boot. When the jumper is installed, NRMMOD is 1 and the console should execute manufacturing tests.

SIR<9> This follows the state of the interrupt from the 53C94 chip. This interrupt indicates that the transfer is complete.

SIR<8> This follows the state of the interrupt from the LANCE.

SIR<7>Floppy HDSTATUS This signal along with SIR<4> follows the state of the floppy density ID bits. SIR<7> SIR<4> = 00 (Low density 1 Mbyte). SIR<7> SIR<4> = 10 (High density 2 Mbytes). SIR<7> SIR<4> = 01 (Extra density 4 Mbytes). SIR<7> SIR<4> = 11 (Drive is not attached).

SIR<6> This interrupt is generated by the SCC that controls the serial port.

SIR<5> This bit reflects the state of the TURBOchannel slot 1 interrupt. A 1 indicates an interrupting condition.

SIR<4>Floppy XDSTATUS This signal along with SIR<4> follows the state of the floppy density ID bits. SIR<7> SIR<4> = 00 (Low density 1 Mbyte). SIR<7> SIR<4> = 10 (High density 2 Mbytes). SIR<7> SIR<4> = 01 (Extra density 4 Mbytes). SIR<7> SIR<4> = 11 (Drive is not attached).

SIR<3> Video interrupt.

SIR<2> Inverted version of video interrupt.

SIR<1> Desktop bus transmit interrupt asserts when the ODB transmit byte register is empty.

SIR<0> Desktop receive interrupt asserts when the ODB receive register is full.

System Interrupt Mask Register

Table F-10. System Interrupt Mask Register 0xBC040120

Bits	Access	Description
31:0	R/W	Interrupt mask

<31:0> These bits, if 0, mask the corresponding interrupt observable in the SIR. Bit <0> masks SIR<0>, bit <1> masks SIR<1>, and so on. The mask does not prevent an interrupt from showing up in the SIR; it merely keeps the CPU from being interrupted. The interrupt mask is set to 0s on power up, masking all interrupts. Software must set to 1 those interrupts that it wants enabled.

Memory Error Register (MER)

The MER stores error information so that a bad memory card can be identified and replaced. Byte masks on writes to this register have no effect. The entire register is always written as a word.

This register is visible in memory space and differs from all other system registers in one respect. It is read from and written to via DMA accesses rather than I/O accesses. Options using the DMA feature of the TURBOchannel can read the MER and obtain some error information.

Table F-11. Memory Error Register 0xAC400000

Bits	Access	Description
31:28	R/W	Reserved
27	R/W	INTR (10ms/1ms Interrupt)
26:17		Reserved
16	R/W	Page boundary error
15	R/W	Transfer length error
14	R/W	PARDIS memory error disable
13:12		Reserved
11:8	R/W	Last byte error
7:0		Reserved

MER<31:28> These bits are reserved.

MER<27> This bit is set every 1ms or 10ms and is intended as a periodic interrupt. When the interrupt is serviced, this interrupt may be cleared by writing this bit to a 0. This interrupt is connected to SIR<12> and is intended to be used for ACE compliance.

MER<16> This bit is set when any memory access crosses a 2-Kbyte boundary.

MER<15> This bit is set when any memory access exceeds 128 words in length.

MER<14> PARDIS disables parity errors when written to 1. This register disables loading of MER<11:8>.

MER<13:12> These bits are reserved.

MER<11:8> This field indicates which byte or bytes had a parity error during a memory read. When set to 1, MER<11> indicates that memory data byte <31:24> is in error, MER<10> corresponds to memory data byte <23:16>, and so on. These bits are set on any memory access where an error occurred and remain set until they are cleared by writing them to 0. MER<14> disables loading this register via memory error.

MSR<7:0> These bits are reserved.

Memory Size Register (MSR)

The MSR supplies memory configuration information. This should be set prior to a memory access (if possible).

Table F-12. Memory Size Register 0xAC800000

Bits	Access	Description
31:27		Reserved
26		INTREN
25		MS10EN
24:21		PFORCE
20		MABEN
19:17		LASTBANK
16:14		Reserved
13	R/W	Size
12:0		Reserved

MSR<31:27> These bits are reserved.

MSR<26>INTREN Setting this bit causes the memory subsystem to generate a periodic interrupt, setting the INTR bit. When INTREN is a 0 no interrupt will be generated.

MSR<25>MS10EN Setting this bit to a 1 sets the period of the interrupt to 10ms. If the bit is 0, then the interrupt period is 1ms.

MSR<24:21>PFORCE This field is used to force parity errors. If PFORCE<0> is set any subsequent memory read accesses will cause a TURBOchannel parity error on byte 0. Setting PFORCE<1> (MSR<22>) will force a parity error on byte 1 etc.

MSR<20>MABEN MABEN enables the A side addresses to the video frame buffer independant of the SIZE bit. This bit should be set to a 1.

MSR<19:17>LASTBANK LASTBANK places the 8 Mbytes of baseboard RAM at a particular location. When set to 0x1, 0x2, 0x3 the baseboard memory starts at bank 0, bank 1 or bank 2 respectively. For example if a SIMM is placed in bank 1 then the baseboard memory should be placed at bank 2 (LASTBANK=0x2). This field is reset to 0 on power up.

MSR<16:14> These bits are reserved.

MSR<13> SIZE is set to a 0 when reset causing the use of 4-Mbyte banks. If SIZE is set to a 1 the memory can utilize 16-Mbyte banks. Different sized memory SIMMs when mixed in a system will not provide optimal storage space.

MSR<12:0> These bits are reserved.

Error Address Register (EAR)

The EAR is a 32-bit register containing a 27-bit word address. Bits <31:29> and <1:0> are zero.

The error address register is set by a memory read parity error, a transfer length or page-crossing error, or an I/O slot read or write timeout.

During either of the above cycles, the error address register is written with the address being referenced. For a cached memory read parity error, this address is the start-of-block address, that is, four-word aligned; otherwise, it is the word address.

The register may be read at any time. CPU writes to this register are ignored. There is no way of checking the validity of this register. Its contents are overwritten each time there is a parity error or a TURBOchannel timeout.

Index

A

Adapter, serial line, 10–5
Addresses
 memory, 12–11
Antistatic kit, 10–2

B

BadVAddr register, F–13
Base system
 Ethernet controller, 12–21
Base system module, 12–21
 clear-NVR connector, 12–21
 CPU module, 12–2
 CPU module connector, 12–21
 diagnostic ROM, 12–21
 ESAR chip, 12–19, 12–21
 install, 12–25
 keyboard-mouse connector, 12–21
 memory modules, 12–7
 option module connectors, 12–21
 power connectors, 12–21
 RAM, 12–21
 remove, 12–23
 SCSI controller, 12–21
 serial port (RS-232), 12–21
 SIMM connectors, 12–21
 TURBOCHANNEL option module
 connectors, 12–21

Base system module tests, E–5
Basic hardware, 1–4

C

Clear-NVR jumper, 1–11

Commands

 boot, D–8
 cat, D–11
 cnfg, D–12
 d, D–21
 e, D–23
 erl, D–25
 go, D–25
 init, D–26
 passwd, D–28
 printenv, D–29
 restart, D–29
 script, D–30
 setenv, D–31
 sh, D–35
 t, D–36
 test, D–36
 uerf, 7–1
 unsetenv, D–37
 xxx, D–7

Commands:

 ls, D–27

- Comm-line-to-MMJ adapter (*see* serial line adapter), 10–5
- Connector pin assignments, C–1 to C–9
- Connectors, 1–8, 1–9
 - CPU module, 1–9
 - Ethernet, 1–8
 - external, 1–8
 - internal, 1–9
 - keyboard-mouse, 1–8
 - power, 1–9
 - external, 1–8
 - RS-232, 1–8
 - SCSI, 1–8, 1–9
 - serial communications (*see* Connectors, RS-232), 1–8
 - SIMM, 1–9
 - TURBOchannel option module, 1–9
 - external, 1–8
- Console commands
 - key combinations, D–2
- Console Commands, D–1
- Control key combinations, D–2
- Controls and indicators, 1–7
- Conventions used in the manual, xxiv
- CPU module, 2–1, 12–2
- CPU registers, F–2
- CPU-type utility, E–33

D

- Data buffers, F–16
- Diagnostic LED, 1–7
- Diagnostic LED codes, E–67
- Drives
 - internal, 11–4

E

- equipment specifications, A–1
 - LK501-AA keyboard, A–9
 - RX26 diskette drive, A–21
 - RZ23L hard disk drive, A–13
 - RZ24L hard disk drive, A–15
 - RZ25 hard disk drive, A–17
 - RZ26 hard disk drive, A–19

- equipment specifications (cont'd)
 - system unit, A–2
 - TLZ06 cassette tape drive, A–21
 - VRC16 monitor, A–4
 - VRM17 monitor, A–6, A–8
 - VSXXX-BB mouse, A–11
- Error address register, F–29
- Error and status register error logs, 7–9
- Error logs, 7–1
 - error and status register, 7–9
 - event types, 7–4
 - format, 7–1
 - memory, 7–7
- Error messages
 - console command;, D–38
- ESAR chip, 12–17, 12–19
 - base system, 12–19
 - option module, 12–17
- Ethernet
 - base system ESAR chip, 12–19
 - option module ESAR chip, 12–17
- Ethernet loopback connector, 10–8
- ?EV, D–38
- ?EVV;, D–38
- Exception program counter register, F–7

F

- Floppy disk drive, 11–6
- Floppy disk drive shield, 11–6
- Floppy disk mounting bracket, 11–10

H

- Halt button, 1–7
- Hardware
 - basic, 1–4
 - configurations, 1–2

I

- Indicator light, 1–7
- Individual tests, E–1

Input devices, 13-1
 mouse, 13-3
 tablet, 13-3
Input devices: keyboard, 13-2
Internal drives, 11-4
 cables, 11-4
 connectors, 11-4
 floppy disk drive, 11-6
 hard disk drive, 11-12
?IO, D-38

K

Keyboard, 2-3, 13-2, B-4

L

LED
 status indicator, 12-26
Loopback connector
 Ethernet, 10-8
 serial line, 10-9
 ThickWire, 10-8

M

Memory
 address ranges, 12-11
 slots, 12-11
Memory addresses, 12-11
Memory error logs, 7-7
Memory error register, F-26
Memory modules, 12-7
 address ranges, 12-11
 in cnfg displays, D-15
 remove and install, 12-11
Memory size register, F-28
Memory slot numbers, 12-11
Memory slots, 12-7
MMJ loopback connector (*see* serial line
 loopback connector), 10-9
Monitors, 2-2
Mouse, 2-3, 13-3, B-4

Multiscreen, 2-2
mxm, 12-21

O

On/off switch, 1-7
Option module
 ESAR chip, 12-17
Option modules
 remove and install, 12-15
Options, 2-1

P

Part numbers, B-1
 adapters, B-5
 basic system components, B-2
 bezels, B-5
 cables, B-4
 connectors, B-5
 cords, B-4
 Ethernet, B-4, B-5
 Floppy disk hardware, B-3
 hardware documentation, B-6
 input devices, B-4
 internal drives, B-3
 keyboard, B-4
 major FRUs, B-2
 monitors, B-4
 mouse, B-4
 power supply, B-2
 SCSI, B-4, B-5
 SCSI hardware, B-3
 small hardware, B-5
 software documentation, B-8
 standoffs, B-5
 system module, B-2
 system unit chassis, B-2
 system unit cover, B-2
 terminators, B-5
?PDE3;, D-38
Pin assignments
 Ethernet loopback connector, Part
 #12-22196-02, C-10
 I/O cover SCSI cable, C-2

Pin assignments (cont'd)

- modem loopback connector, Part#12-15336-13, Option #H3200, C-10
 - printer/communications connector, C-4
 - printer/console loopback connector, C-6
 - SERIAL.bus, C-8
 - Sound Connector Pin Assignments, C-9
 - summary of loopback connections, C-10
 - ThickWire Ethernet connector, C-3
- Power, 1-7
- Power supply, 12-26

R

Registers, F-2

- BadVAddr, F-13
 - cause, F-3
 - CPU, F-2
 - data buffers, F-16
 - EAR, F-29
 - (EPC), F-7
 - ER, F-26
 - error address, F-29
 - exception program counter, F-7
 - memory error, F-26
 - memory size, F-28
 - MSR, F-28
 - SIR, F-20
 - SSR, F-17
 - status, F-8
 - system, F-15
 - system interrupt, F-20
 - system interrupt mask, F-26
 - system support, F-17
 - xxxx, F-20
- RX26 floppy disk drive, 11-26
- RZ23L hard disk drive, 11-15
- RZ24 hard disk drive, 11-17
- RZ24L hard disk drive, 11-19

RZ25 hard disk drive, 11-20

RZ26 hard disk drive, 11-23

S

SCSI

- cable lengths, 11-3
 - cables, 2-4
 - controllers, 2-4
 - drives, 2-4
 - external, 2-4
 - internal, 2-4
- SCSI chain terminator, 10-7
- SCSI controller terminator, 10-6
- Serial line adapter, 10-5
- Serial line loopback connector, 10-9
- SIMM addresses, 12-11
- SIMMs, 12-7
 - remove and install, 12-11
- Slot numbers
 - memory, 12-11
- ?SNF, D-38
- Speaker, 12-26
- Staticsensitive materials, 10-2
- Status register, F-8
- ?STF, D-38
- ?STX, D-38
- System Boot ROM switch, 1-12
- System interrupt mask register, F-26
- System interrupt register, F-20
- System module, 12-21
- System support register, F-17
- System unit cover, 10-3

T

Tablet, 13-3, B-4

Terminal, 2-2

Terminator

- SCSI chain, 10-7
- SCSI controller, 10-6

Test

- diagnostics, E-62

Tests

- ab/caps, E-6
- ab/cntl, E-7
- ab/ctrs, E-7
- ab/nm, E-8
- ab/t, E-8
- base system module, E-5
- cache/data, E-8
- cache/fill, E-9
- cache/isol, E-11
- cache isolate, E-11
- cache/reload, E-12
- cache/seg, E-13
- collision, E-34
- common diagnostic utilities, E-35
- crt/ascii, E-15
- crt/cursmem, E-16
- crt/e, E-16
- crt/fb, E-16
- crt/h, E-17
- crt/init, E-17
- crt/patterns, E-18
- crt/vdac, E-18
- cyclic redundancy code (CRC), E-36
- desktop bus controller, E-7
- desktop bus tables verification tool,
E-8
- display next message, E-8
- DMA registers, E-37
- DMA transfer, E-39
- epm/dump, E-18
- epm/init, E-20
- epm/sign, E-19
- epm/write, E-20
- ESAR, E-40
- Ethernet DMA registers, E-37
- Ethernet DMA transfer, E-39
- Ethernet station address ROM
(ESAR), E-40
- external loopback, E-41
- fctr/intvl, E-20
- fctr/run, E-21
- fdi/dor, E-21
- fdi/exfdc, E-22

Tests (cont'd)

- fdi/read, E-23
- fdi/reset, E-24
- fdi/t, E-25
- fdi/wrtrd, E-25
- floating I/O memory, E-30
- floating-point unit, E-27
- fpu, E-27
- internal loopback, E-43
- interrupt request (IRQ), E-42
- IRQ, E-42
- isdn/audio, E-28
- isdn/base, E-29
- ISDN baseline, E-29
- isdn/dma, E-30
- LANCE, E-35
- LANCE setup utilities, E-48
- mem, E-31
- mem/float10, E-30
- mem/init, E-32
- memory module, E-31
- mem/select, E-32
- mfg/done, E-33
- misc/cpu-type, E-33
- misc/wbpart, E-33
- multicast, E-44
- ni/cllsn, E-34
- ni/common, E-35
- ni/crc, E-36
- ni/dma1, E-37
- ni/dma2, E-39
- ni/esar, E-40
- ni/ext-lb, E-41
- ni/int, E-42
- ni/int-lb, E-43
- ni/m-cst, E-44
- ni/promisc, E-46
- ni/regs, E-47
- ni/setup, E-48
- nonvolatile RAM (NVR), E-49
- NVR, E-49
- partial write, E-33
- promiscuous mode, E-46
- RAM select lines, E-32

Tests (cont'd)

- real time, E-51
- real-time clock (RTC) period, E-50
- real-time clock (RTC) registers, E-50
- registers, E-47
- rtc/nvr, E-49
- rtc/period, E-50
- RTC period, E-50
- RTC registers, E-50
- rtc/regs, E-50
- rtc/time, E-51
- scache/data, E-15
- scc/access, E-52
- SCC access, E-52
- SCC direct memory access, E-53
- scc/dma, E-53
- scc/int, E-55
- SCC interrupts, E-55
- scc/io, E-55
- SCC i/o, E-55
- scc/pins, E-57
- SCC pins, E-57
- SCC transmit-receive, E-59
- scc/tx-rx, E-59
- scsi/cntl, E-61
- SCSI controller, E-61
- scsi/sdiag, E-62
- SCSI send diagnostics, E-62
- scsi/target, E-63
- SCSI target, E-63
- serial communication chip (SCC)
 - direct memory access, E-53
- serial communication chip (SCC)
 - interrupts, E-55
- serial communication chip (SCC) i/o, E-55
- serial communication chip (SCC) pins, E-57
- serial communication chip (SCC)
 - transmit-receive, E-59
- serial communications chip (SCC)
 - access, E-52
- tlb/prb, E-66
- TLB probe, E-66

Tests (cont'd)

- tlb/reg, E-66
- TLB registers, E-66
- translation lookaside buffer (TLB)
 - probe, E-66
- translation lookaside buffers (TLB)
 - registers, E-66
- zero memory, E-32

Tests:

- ASCII font, E-15
- cache segment, E-13
- chip initialization, E-17
- cursor memory, E-16
- desktop bus counters, E-7
- desktop bus device capabilities, E-6
- DIAGDN bit, E-33
- 'E', E-16
- EEPROM contents display, E-18
- EEPROM read/modify/write, E-20
- EEPROM reinitialization, E-20
- EEPROM signature data check, E-19
- exercise floppy disk controller, E-22
- floppy disk debug tools, E-25
- floppy DOR register test, E-21
- floppy media read test, E-23
- frame buffer, E-16
- free-running counter increment, E-21
- free-running counter interval, E-20
- 'H', E-17
- ISDN controller audio, E-28
- ISDN controller peripheral port, E-30
- reset floppy disk interface, E-24
- scache data, E-15
- screen patterns, E-18
- vdac/color palette, E-18
- write/read floppy media, E-25

Tests:individual, E-1

?TFL, D-38

ThickWire loopback connector, 10-8

TLZ06 cassette tape drive, 11-25

?TNF, D-38

TURBOchannel option modules, 2-4
 remove and install, 12-15

?TXT, D-38

U

ULTRIX error logs, 7-1

Utilities

 display MOP counters, E-37

 ni/ctrs, E-37